End User Documentation **CUSTOMER**

Document Version: 1.0 - 2014-11

SAP InfiniteInsight® Explorer – Analytical Data Management v7.0

User Guide

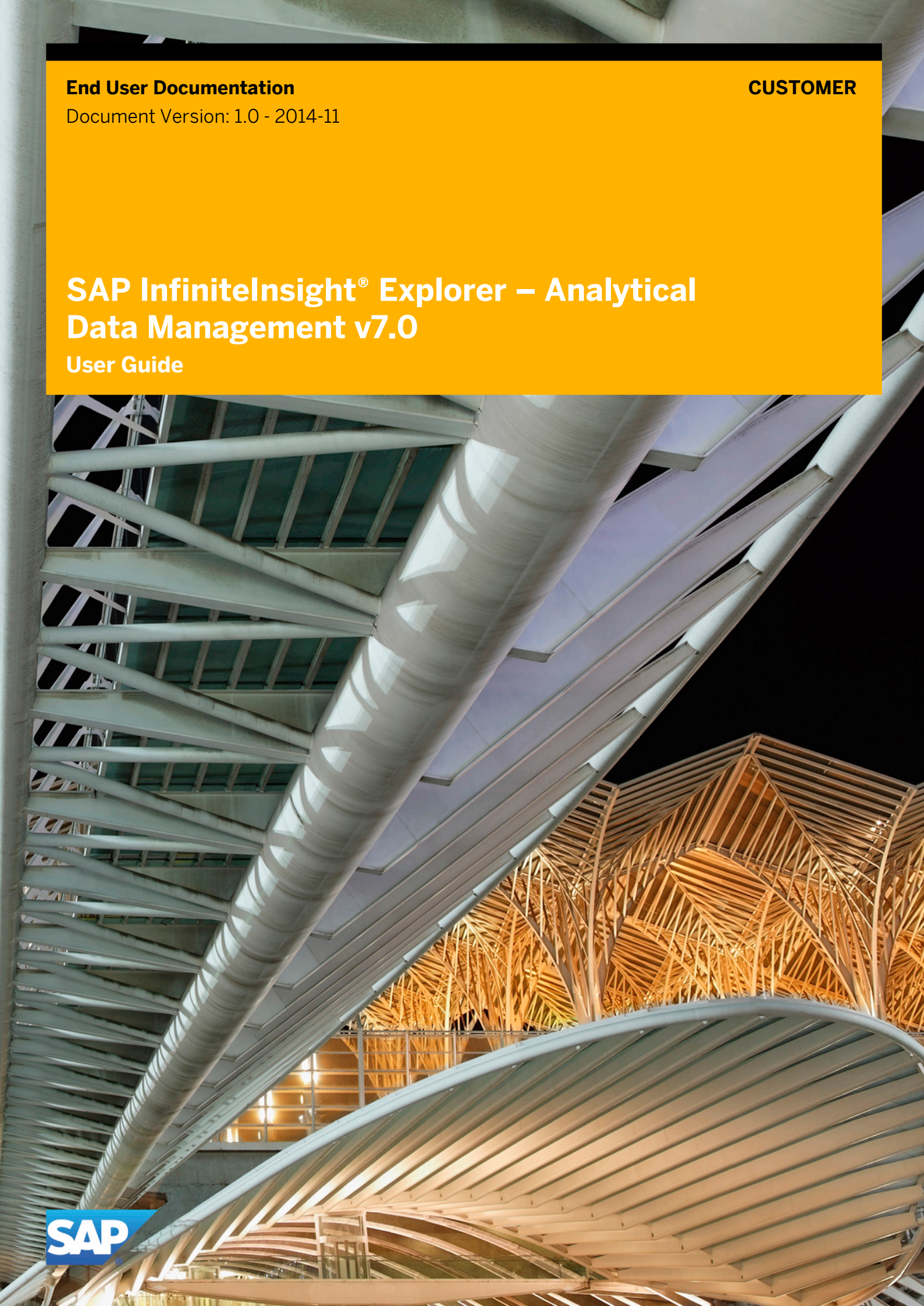


Table of Contents

1 Welcome to this Guide 3
1.1 What this Document Covers 3

2 Essential Concepts 4
2.1 Definitions 4
2.1.1 InfiniteInsight® Explorer - Semantic Layer 4
2.1.2 Entity 4
2.1.3 Time-stamped Population 5
2.1.4 Analytical Record 5
2.1.5 Analytical Data Set 5
2.1.6 Temporal Analytical Data Set 5
2.1.7 Performance Indicator (PI) 6
2.2 Methodology 7

3 Application Scenario: Segmented Cross-sell in Retail Banking 8
3.1 Presentation 8
3.2 Creating the Performance Indicator 8
3.2.1 Creating the Calendar Table 9
3.2.2 Creating the Performance Indicator 10
3.3 Preparing the Data 18
3.3.1 Creating the Entity 18
3.3.2 Creating the Analytical Record 22
3.3.3 Updating the Default Time-stamped Population 28
3.3.4 Copying Objects to Another Metadata Repository 42
3.4 Creating the First Classification Model 44
3.4.1 Selecting the Data to be Modeled 45
3.4.2 Describing the Data 49
3.4.3 Selecting the Variables 51
3.4.4 Checking the Model Parameters 53
3.5 Analyzing the Model Results 54

4 Performing an Entity Transfer 55
4.1 Transfer an Entity 55
4.2 Use the Transferred Entity in a New Database 59

© 2014 SAP SE or an SAP affiliate company. All rights reserved-

1 Welcome to this Guide

1.1 What this Document Covers

This document presents the new data manipulation functions offered by SAP InfiniteInsight® through a scenario. The first section provides definitions of essential concepts as well as the methodology to be applied when using these new functions. The second section guides you through the steps defined in the methodology by using a practical scenario as an example.

 © 2014 SAP SE or an SAP affiliate company. All rights reserved-

2 Essential Concepts

IN THIS CHAPTER

Definitions 4
Methodology 7

2.1 Definitions

2.1.1 InfiniteInsight® Explorer - Semantic Layer

The **InfiniteInsight® Explorer - Semantic Layer** as defined by SAP InfiniteInsight® is made of three elements:

- the **Data Manipulation features** offered by SAP InfiniteInsight®, such as filters, join attributes, new attributes computation, aggregates, performance indicators definition,
- the SAP InfiniteInsight® **analytical data set methodology** (see Methodology (on page 7)),
- the **metadata management**, which allows you to store, share and easily re-use the data descriptions.

2.1.2 Entity

An **entity** is the object of interest of any analytical task. It can be a customer, a product, a store, and is usually identified with a single identifier that can be used throughout the data repositories. Entities are usually associated with a state model describing the life cycle of such an analytical object of interest.

Note

This is a technical constraint: entities MUST be uniquely identified.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-

2.1.3 Time-stamped Population

A **time-stamped population** is a set of entities at a particular point in time. For example all customers as defined by `customer_ID` (the entities) as of January 1, 2008 (the time stamp).

It is a list of pairs <identifiers; time stamps>: the semantic meaning of such a construct can be associated with snapshots of the entities and a given time: in general terms, a given entity may be represented at different time stamps in a single time-stamped population.

2.1.4 Analytical Record

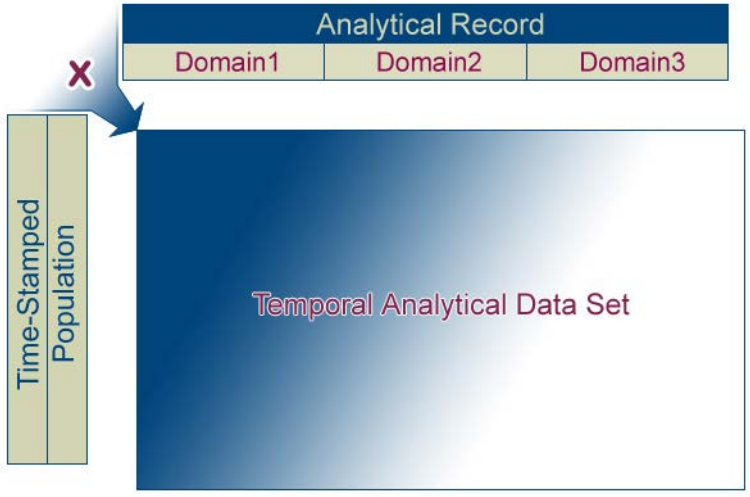
An **analytical record** is a logical view of all **attributes** corresponding to an **entity**. An analytical record may be decomposed into domains that group attributes related to each other. For example, in CRM, an analytical record can have a demographic domain and a behavioral domain.

2.1.5 Analytical Data Set

An analytical data set is a data set generated by manipulating data through merges, creation of new fields, application of filters, and so on.

2.1.6 Temporal Analytical Data Set

A **temporal analytical data set** is a special case of analytical data set. It is the product of a **time-stamped population** by an **analytical record**, the result of this operation can be seen as a virtual table containing attributes values associated with identifiers in relation with the time stamp.



© 2014 SAP SE or an SAP affiliate company. All rights reserved-

In other words, a temporal analytical data set contains 'photos' or snapshots of a given list of entities taken at a given time. This time can be different for each entity, and an entity can be associated with several 'photos'.

Note

The analytical data sets are used to train predictive/descriptive models and to apply these models.

2.1.7 Performance Indicator (PI)

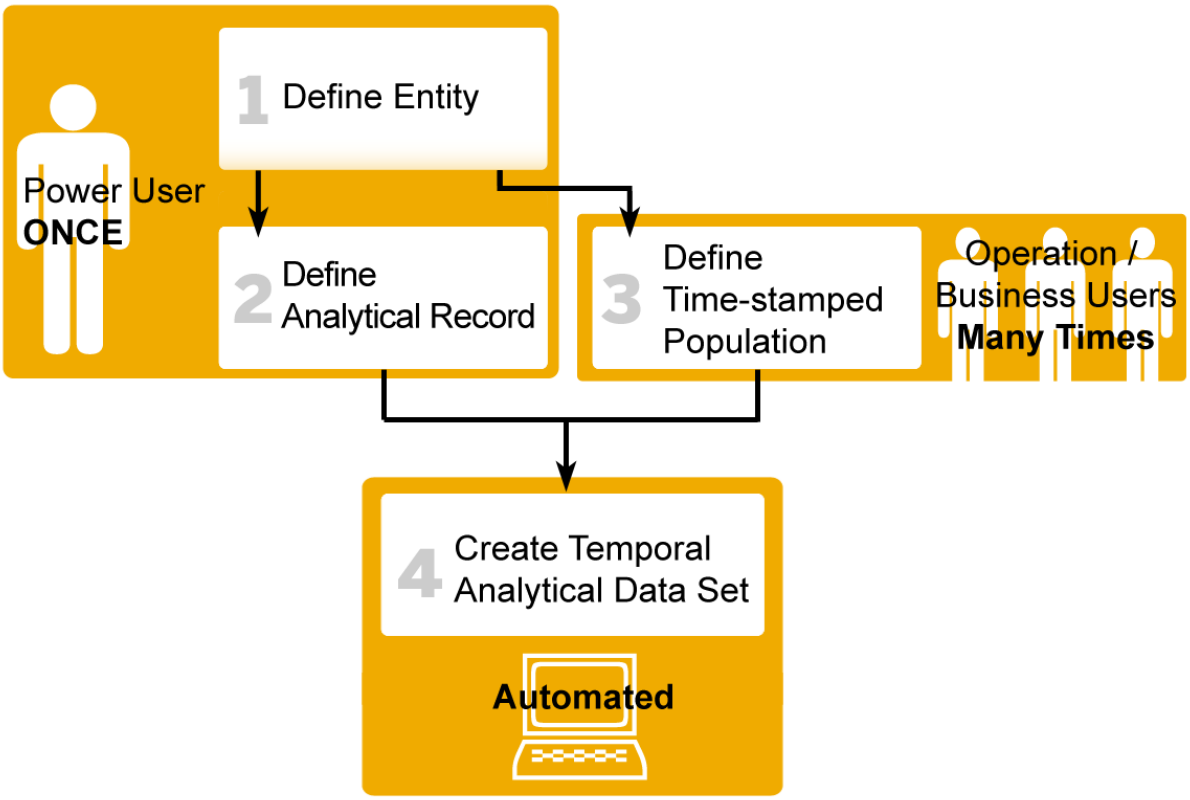
The **Performance Indicators (PIs)** help organizations achieve organizational goals through the definition and measurement of progress. The purpose of defining PIs is to have a common definition of a metric across multiple projects. A metric like "customer value" could easily be defined in several different ways, leading to confusing or contradictory results from one analysis to the next. Shared PIs ensure consistency across analysts and projects over time. The key indicators are agreed upon by an organization and are indicators which can be measured and will reflect success factors. The PIs selected must reflect the organization's goals, they must be the key to its success, and they must be measurable.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-

2.2 Methodology

The methodology used in SAP InfiniteInsight® for InfiniteInsight® Explorer - Semantic Layer has been designed around systematization and automation of the creation of temporal analytical data sets. The key is to define **all** attributes describing a customer through expressions that are not based on basic information only, but also on a **time stamp**. In other words, all attributes are expressions that can be computed in order to get the value of that attribute for a given customer **at a given time**.

This methodology is made of four steps:



1. **First**, you need to **define the entity** that will be the main object of the analytical tasks.
2. **Second**, you need to define the **analytical record** that describes the entity's attributes on which predictive and descriptive models can be trained.

> **Note**
>
> The first two steps only need to be defined **once** by a power user.

3. The **third** step, during which you **define the time-stamped population** to be analyzed, will be executed by **operation or business analysts** and can be redone **several times** to adapt the population to your needs.
4. The **last** step is the **automated** creation of the temporal analytical data set consisting of the product between the analytical record, which determines the number of columns, and the time-stamped population, which determines the number of lines. InfiniteInsight™ also offers the possibility to **create a target on the fly** when it does not exist in the original data.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-

3 Application Scenario: Segmented Cross-sell in Retail Banking

IN THIS CHAPTER

Presentation....8
Creating the Performance Indicator....8
Preparing the Data....18
Creating the First Classification Model....44
Analyzing the Model Results....54

3.1 Presentation

In this scenario, you are a data mining analyst in a large retail bank. Your role is to translate any business question (in this case, cross-sell) into successive data mining functions/models. You are part of a center of excellence in analytics. This group provides support for teams such as the marketing team when developing advanced analytics projects. Your project is to increase the number of credit cards subscribed by present customers.

At the start of year 2008 (January 7th), the marketing management team decided on an objective to increase the number of credit cards subscribed by present customers of the bank. They decided to achieve this objective through the use of advanced analytics. Their goal is to engage the marketing operations of the retail bank on the path to 'competing on analytics'.

3.2 Creating the Performance Indicator

As described in the section Definitions (on page 4) > Performance Indicator (PI) (on page 6), creating a performance indicator will allow you to measure the performance of your company regarding the goal you want to achieve.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

3.2.1 Creating the Calendar Table

Scenario

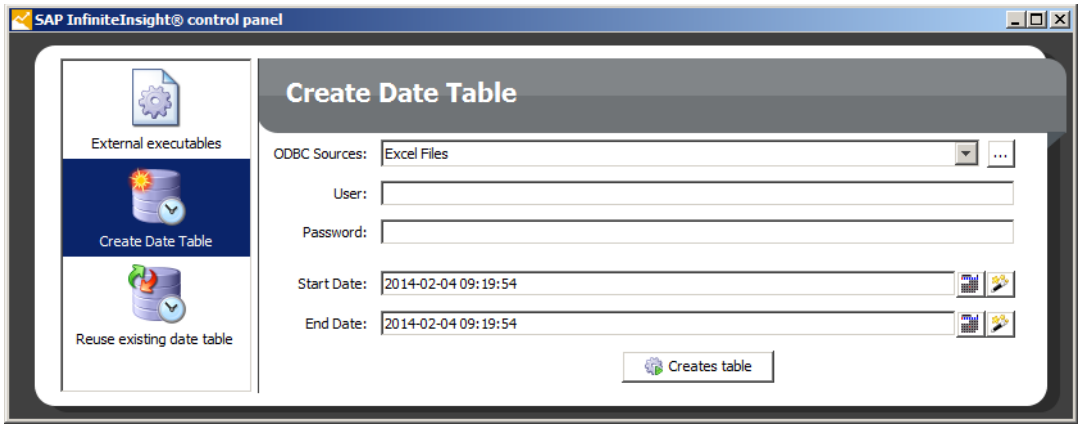
For a given date, the Calendar table allows you to get specific information relative to the date that is complex to compute such as the day of the week, the week number in the year, the quarter and so on. This information is needed to compute the performance indicators.

Settings

- *ODBC Sources*: the database in which you have imported the PKDD data.
- *Start Date*: `1990-01-01 00:00:00`
- *End Date*: `2020-01-01 00:00:00`

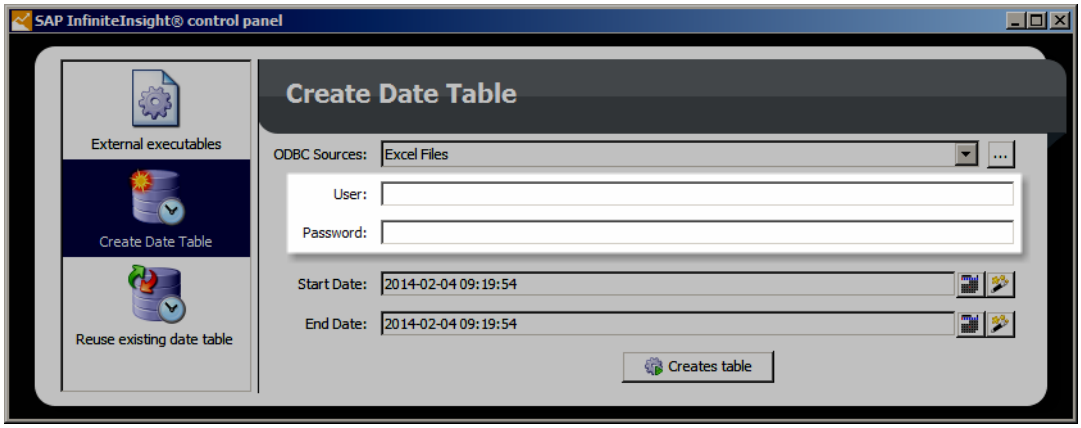
To Create the Calendar Table

1 In the Windows *Start* menu, select the option *Programs* > *SAP Business Intelligence* > *SAP SAP InfiniteInsight®* > *Control Panel*.



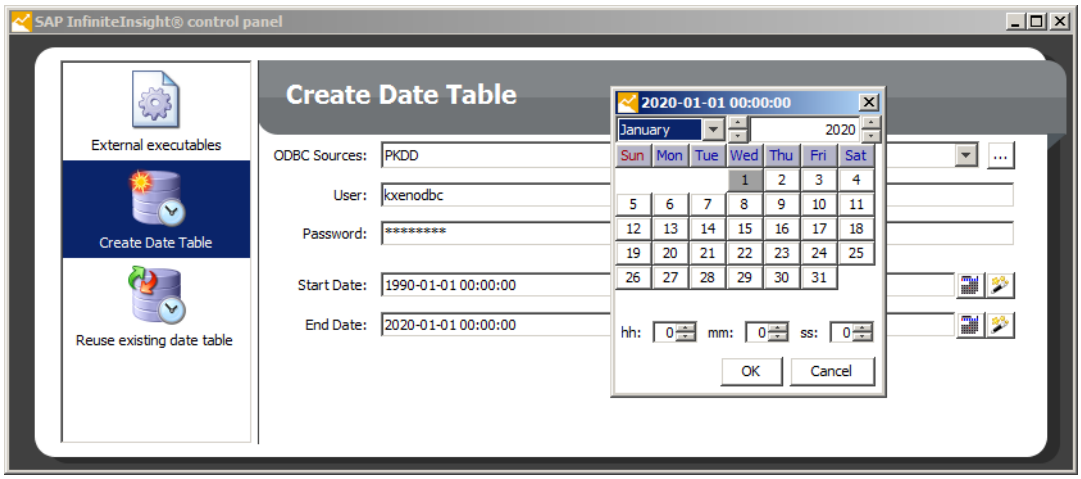
2 In the list *ODBC Sources*, select the database in which you want to create the date look-up table.

3 If the database is password protected, enter the user name and password respectively in the fields *User* and *Password*.



© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

4 Use the fields *Start Date* and *End Date* to define the time range that should be covered by the date look-up table. This time range needs to cover the content of your database.



5 Click the button *Create Table*. A message box is displayed indicating that the table is being created.

6 Close the *Control Panel*.

3.2.2 Creating the Performance Indicator

Scenario

A simple performance indicator can be created to visualize the evolution of the credit cards sales each month. You will need to create an aggregate counting the number of credit cards bought in the past month, and shift it one month forward every month.

Settings

- Shifting Aggregate:
  - *Event Table*: `credit_card`
  - *Date Column*: `card_date`
  - *Function*: `count`
  - *Target Column*: `card_id`
  - *Aggregation Period*: `1 Month in the Past`
  - *Name*: `count_creditcard`
- Period:
  - *Compute for a period of* `730 days` (which is the number of days between August 1, 1996 and August 1, 1998)
  - *Ending at*: `1998-08-01 12:00:00`
- Advanced Filters:
  - Create the condition: `date_day == 1`

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

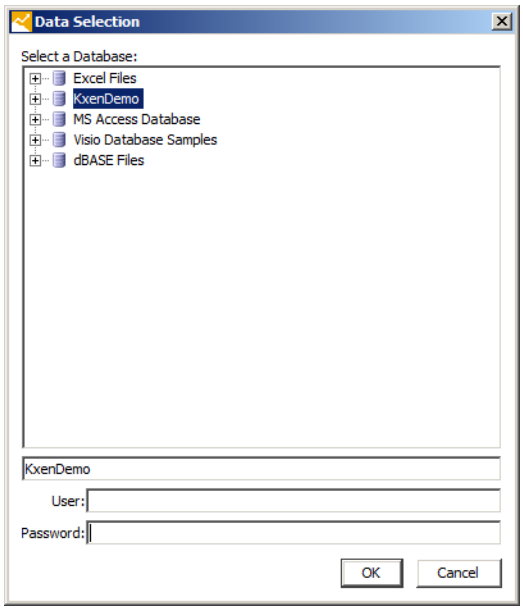
- Performance indicator name: `PI_CountCreditCard`

☑ To Create a Performance Indicator

1 In SAP InfiniteInsight® start panel, select the option *Create or Edit Explorer Objects* in the *Explorer* section.

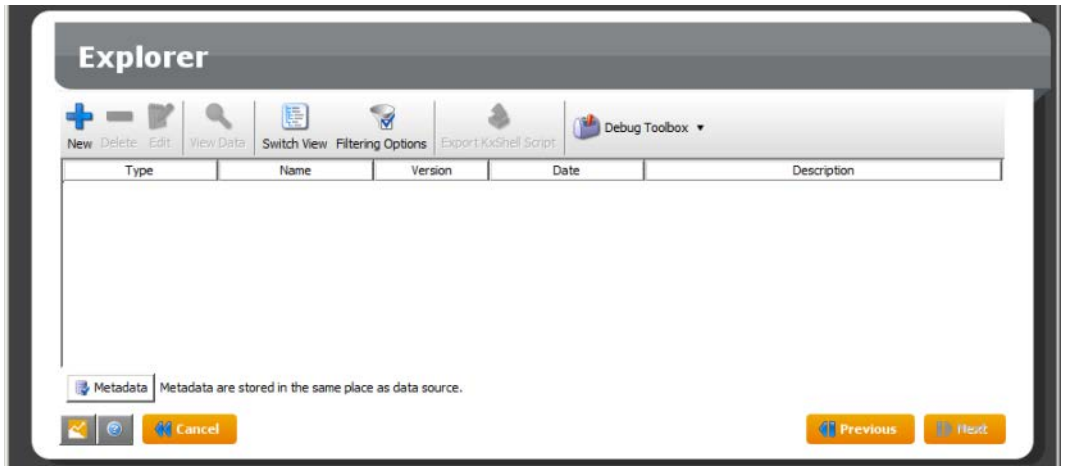
2 Click the *Browse* button to select the database in which your data are stored.

3 If the database is protected, enter the username in the *User* field and the password in the *Password* field.



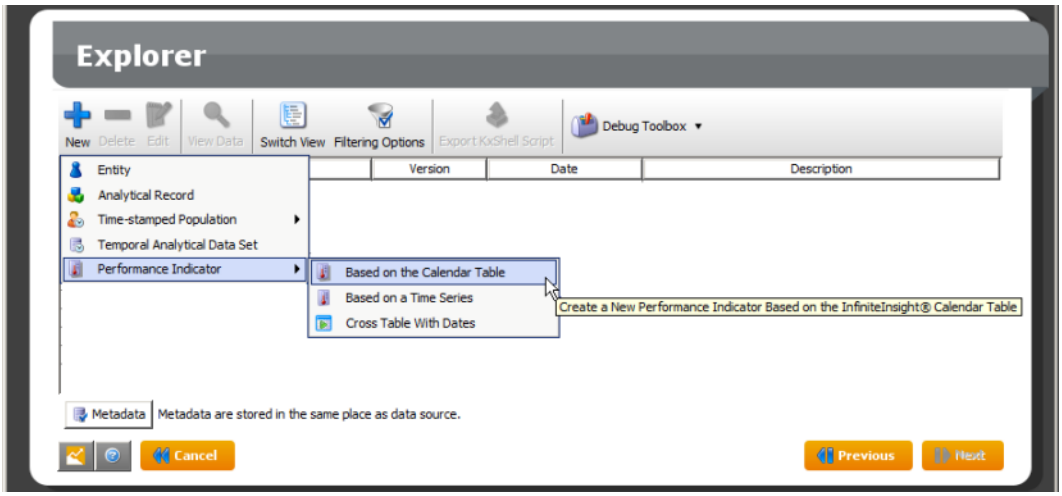
4 Click *OK* to validate your selection.

5 Click *Next* to display the panel *Explorer*.

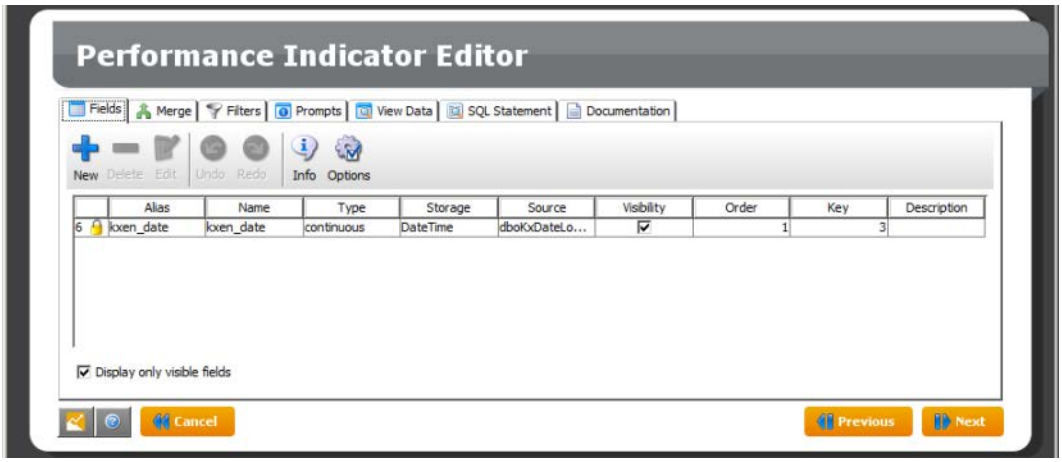


© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

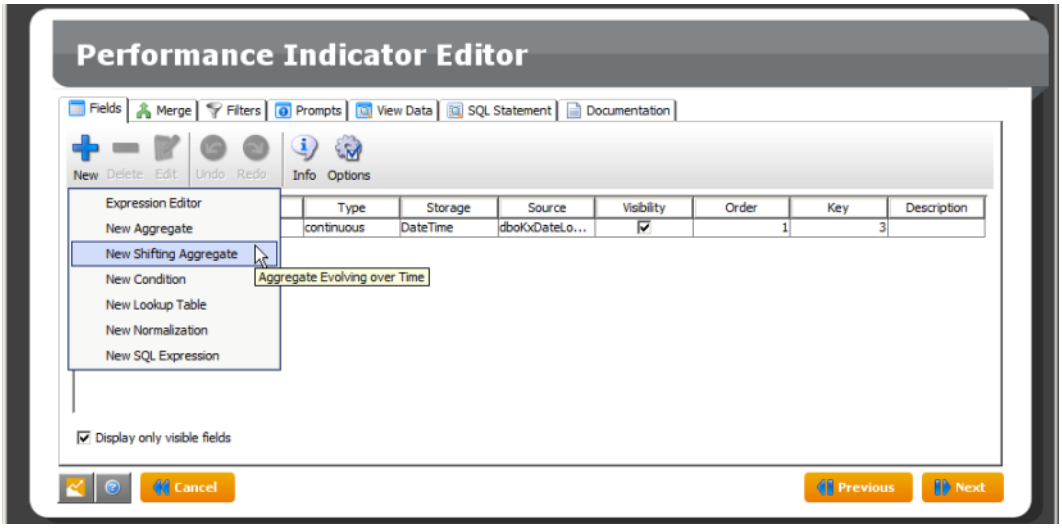
6 In the + menu, select the option *Performance Indicator* > *Based on the Calendar Table*.



The *Performance Indicator Editor* opens.

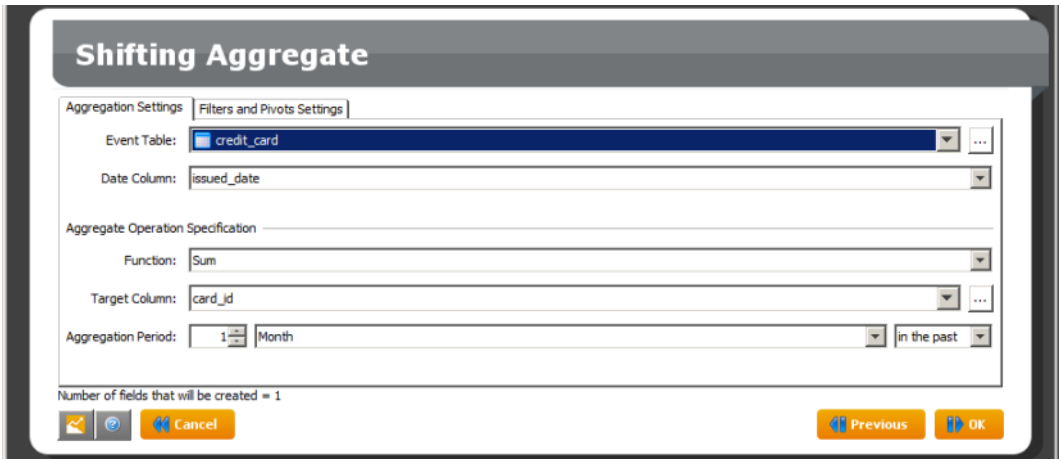


7 In the *Fields* tab, click the + button and select the option *New Shifting Aggregate*.



© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

8 In the list *Event Table*, select the table containing the data on which you want to create the aggregate.



© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

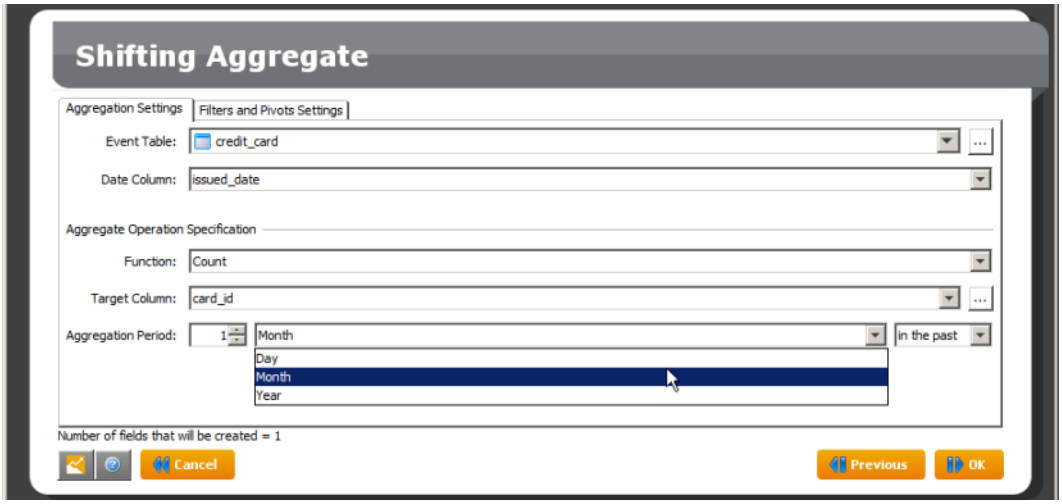
9 In the list *Date Column*, select the column which contains the date the aggregate should be based on.

10 In the *Function* list, select the aggregation operation you want to apply. The following operations are available:

Functions	Description	Returned Values
Count	computes the number of not null values found in the selected column of the Event table **Note** This may differ on some databases and behave as a count (no operand).	number of not null occurrences in the selected column
Count (no operand)	computes the total number of lines in the Event table	total number of occurrences
Sum	compute the sum	sum
Average	compute the mean	mean
Min	identifies the minimum value	minimum value
Max	identifies the maximum value	maximum value

11 In the list *Target Column*, select the column on which you want to apply the selected aggregation function.

12 On the line *Aggregation Period*, select the period on which you want to compute the aggregate.

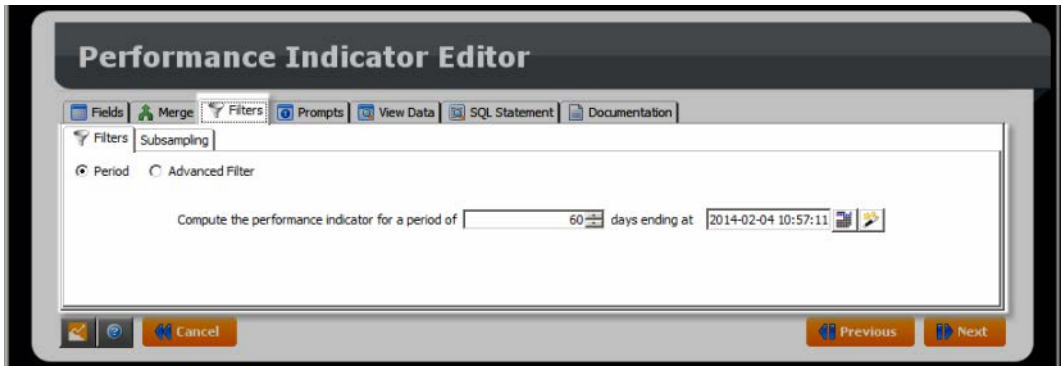


© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

13 Click the *Validate* button. A pop-up window opens allowing you to set a name for the new field you are creating.

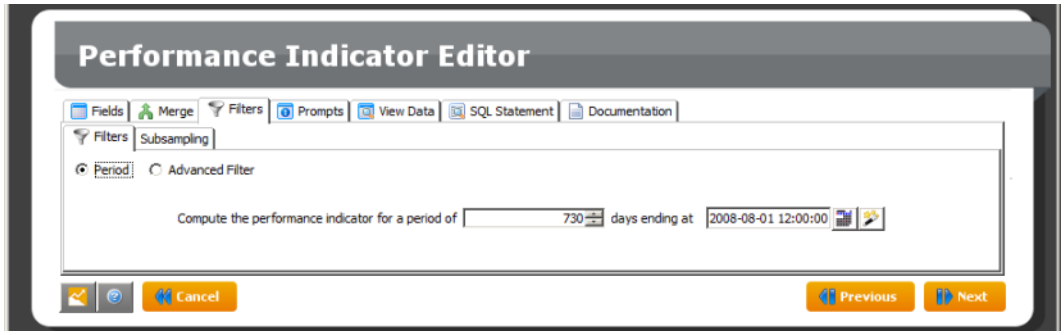
14 In the *Name* field, enter the name you want to give to the field.

15 Click the *Filters* tab.



16 Select the *Period* option.

17 Use the provided fields to select the period over which you want to compute the performance indicator.



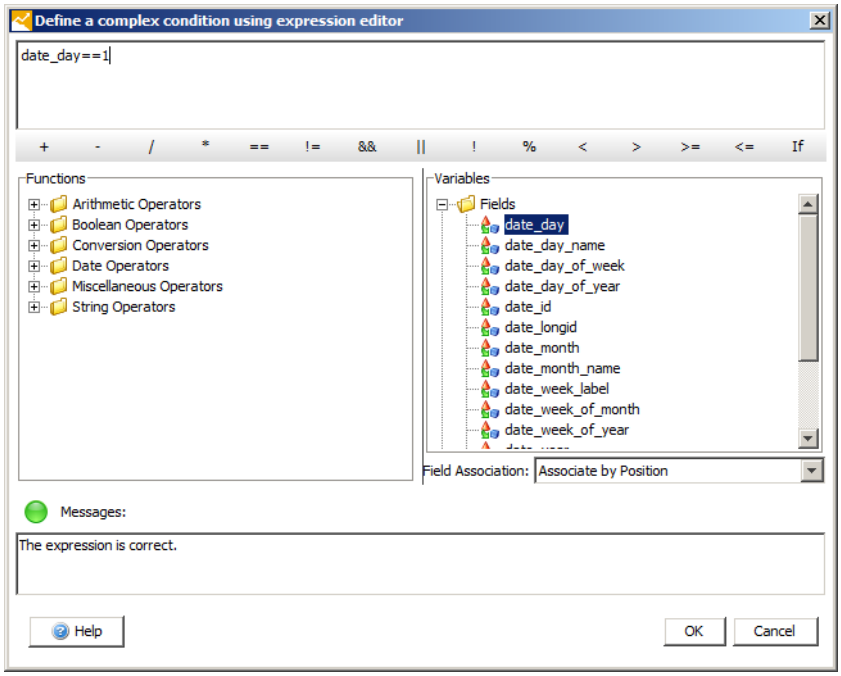
By default, SAP InfiniteInsight® will compute one point for each line in the table used to create the performance indicator, that is, either the Calendar table or the time series you have based your indicator on. If you want to compute only certain points, for example one point every month, you will have to create an advanced filter to keep only the dates that interest you. As an example, if you want a point at the beginning of each month, you will create the filter `date_day == 1`.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

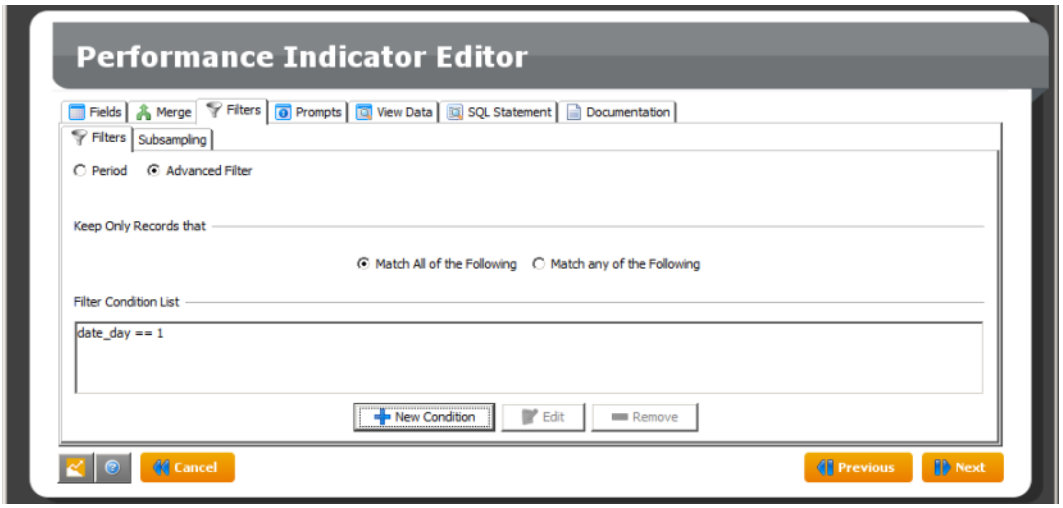
18 Select the option *Advanced Filters*.

19 Click the button *New Condition*. The expression editor opens.

20 Define the filter you want to apply.

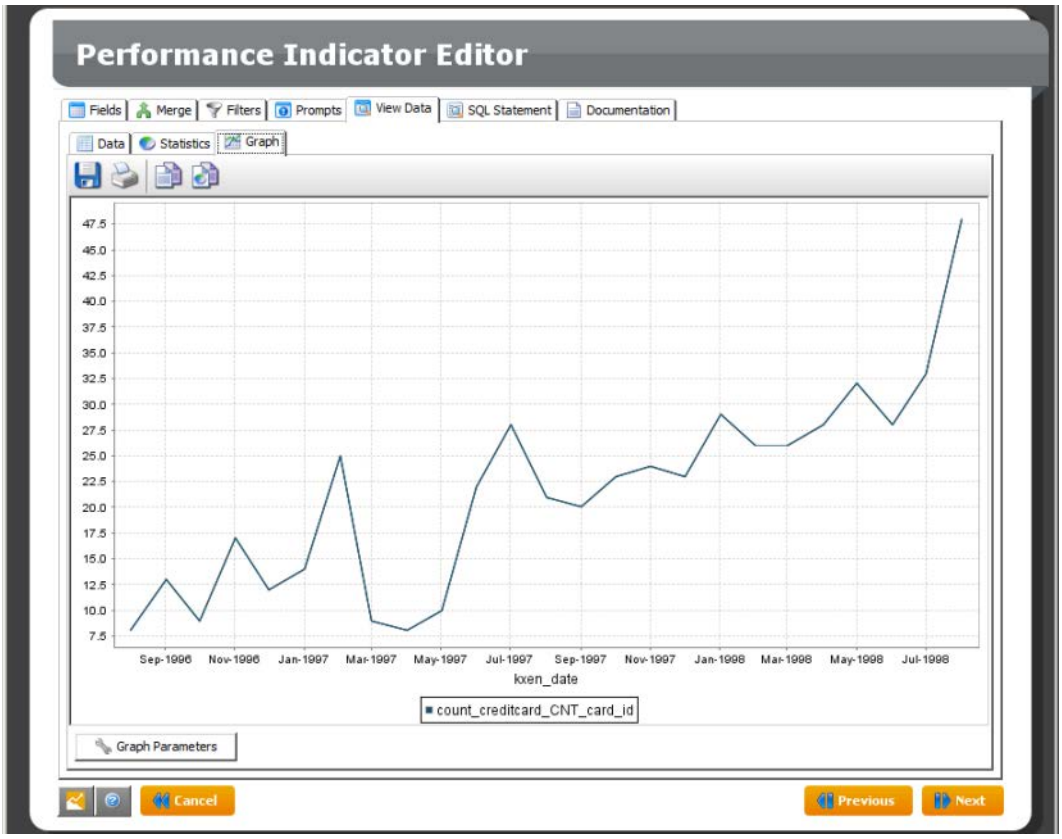


21 Click the *OK* button. The expression editor closes and the defined condition appears in the *Filter Condition List*.

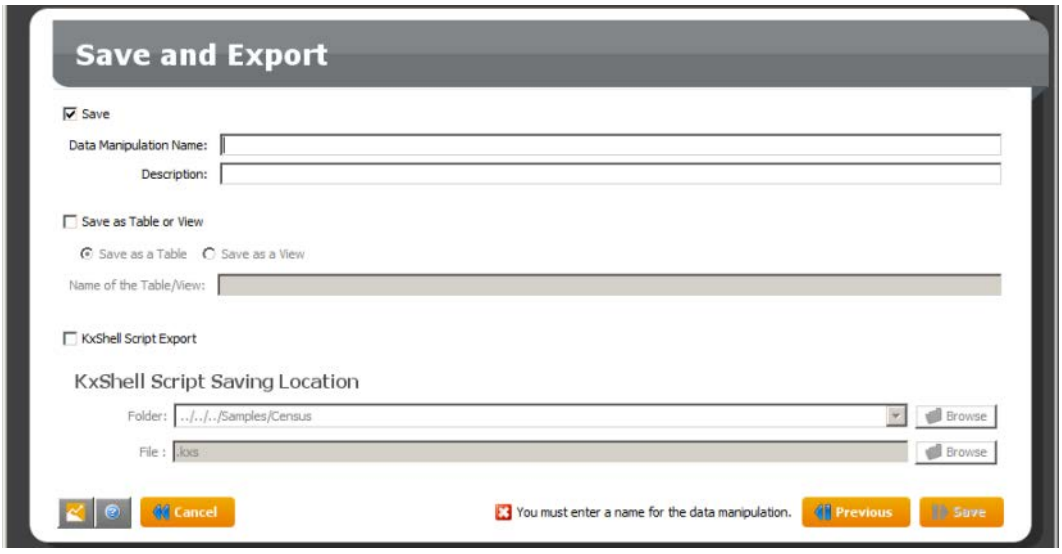


© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

22 Click the tab *View Data* to display the performance indicator as a graph.



23 Click *Next*; the panel *Save and Export* is displayed.



24 In the field *Data Manipulation Name*, enter a name for your performance indicator.

25 Use the *Description* field to enter a description of the performance indicator.

26 Click *Save*.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

3.3 Preparing the Data

3.3.1 Creating the Entity

Scenario

In this particular case, even if the problem statement for the project speaks about selling credit card to '*customer*', since credit cards are associated with accounts, you have to determine, together with everyone involved in this project, if the entity for this project is the '*account*' (on which the credit card is attached), the 'customer' (the only one that can receive a piece of mail describing the offer), or the association between the account and the customer (which is called a '*Disposition*' in the database schema), describing the role of the person with the account. There are two roles in our example: one is the 'owner' of the account, and another possible role is a 'user' of the account.

To define an entity, you must take into account the fact that this entity makes business sense, that it can be characterized in terms of attributes, that it can be associated with predictive metrics in relation with the tasks you want to perform (in this case the number of gold credit cards). Defining an entity is not a minor operation. Entities may be used in many projects and cannot be changed without an impact analysis on all the deployed processes using this entity.

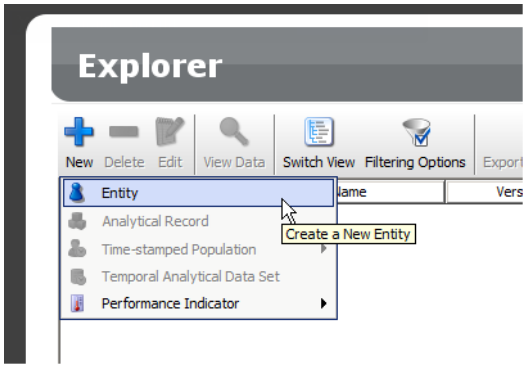
The entity selected is the '*Disposition*'.

Settings to Use

- Enter `entity_disposition` as the *Name of the entity*.
- Select `disposition` as the *Table*.
- Select `disp_id` as the *Id Field*.
- Do not define a *Filter*.

To Create an Entity

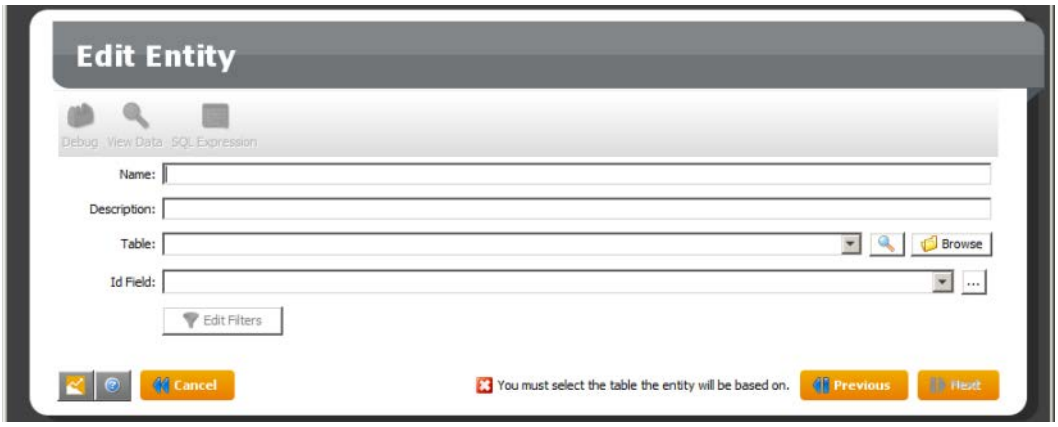
1 In the + menu of the panel *Explorer*, select *Entity*.



The panel *Edit Entity* is displayed.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

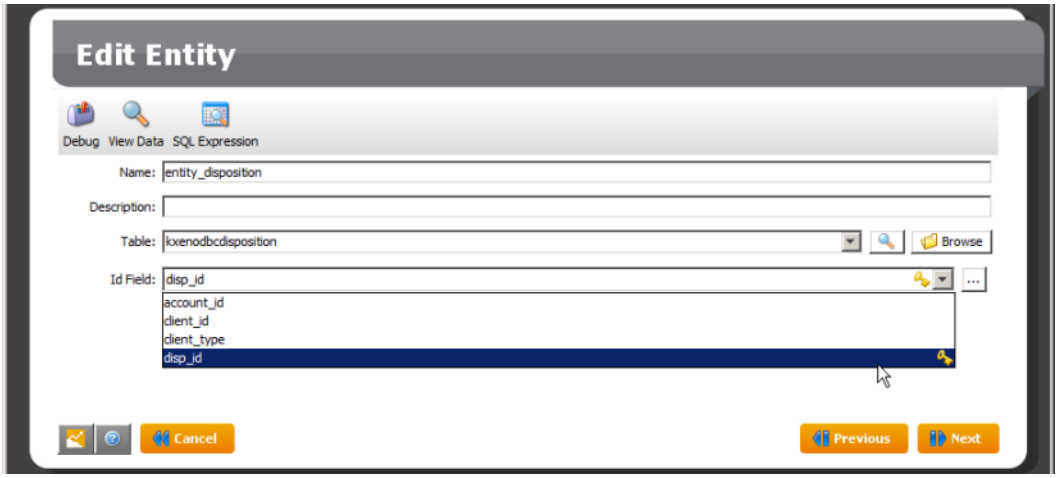
2 In the field *Name*, enter the name of the new entity.



3 In the field *Description*, enter a description of the new entity.

4 To select the table on which you want to build the new entity, click the *Browse* button and select a table in the displayed dialog box. Click *OK* to validate your choice.

5 In the drop-down list *Id Field*, select the column containing the entity identifier.



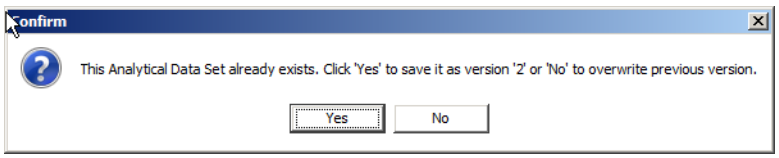
6 *Optional:* to add a filter to the entity in order to remove some lines depending on a specific condition, click the button *Edit Filter*. The existence of a filter is indicated as listed below:

- : no filter has been defined
- : a filter already exists

For the detailed procedure, see To Edit a Filter (on page 20).

7 Click *Next* to validate the creation or changes on the entity.

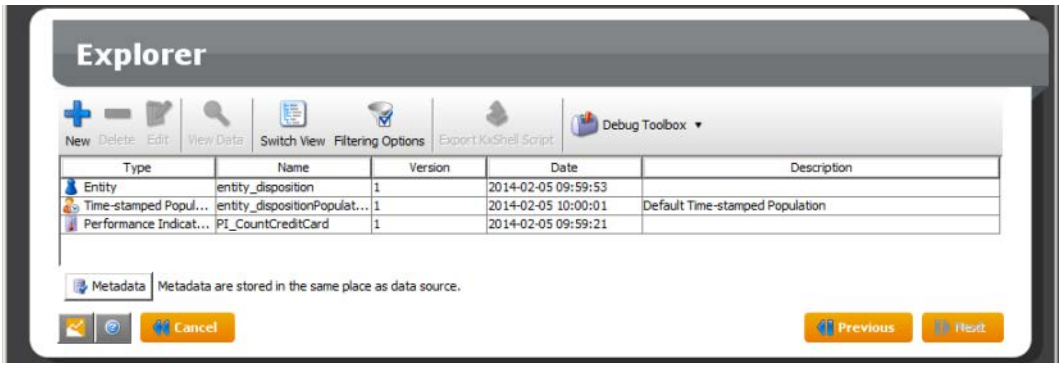
8 If an entity with the same name already exists, a dialog box is displayed asking you to choose what should be done.


Confirm
This Analytical Data Set already exists. Click 'Yes' to save it as version '2' or 'No' to overwrite previous version.
Yes
No

- Click *Yes* to save the entity as a new version of the existing one.
- Click *No* to overwrite the existing entity.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

The panel *Explorer* is displayed, listing the newly created entity and the default time-stamped population associated with it.



Caution

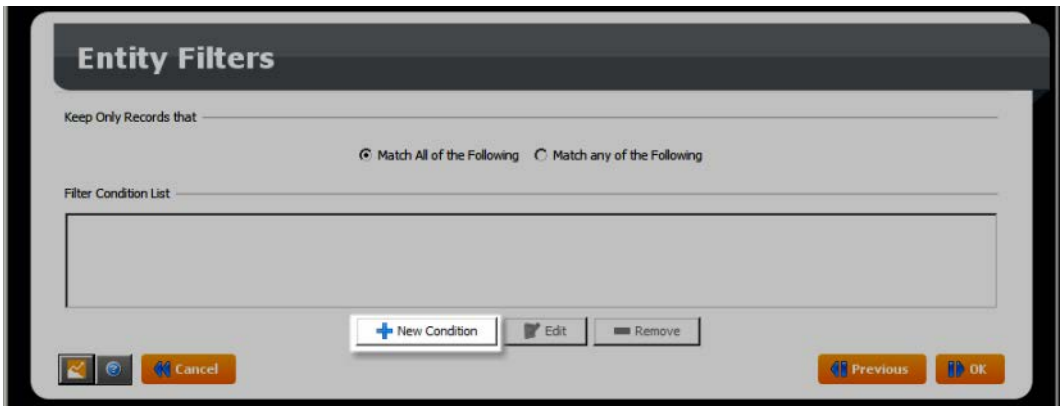
Because of technical constraints, the default time-stamped population must NOT be deleted **as long as the entity exists**.

☑ To Edit a Filter

1 Click the button *Edit Filters*.

The panel allowing you to define filters opens.

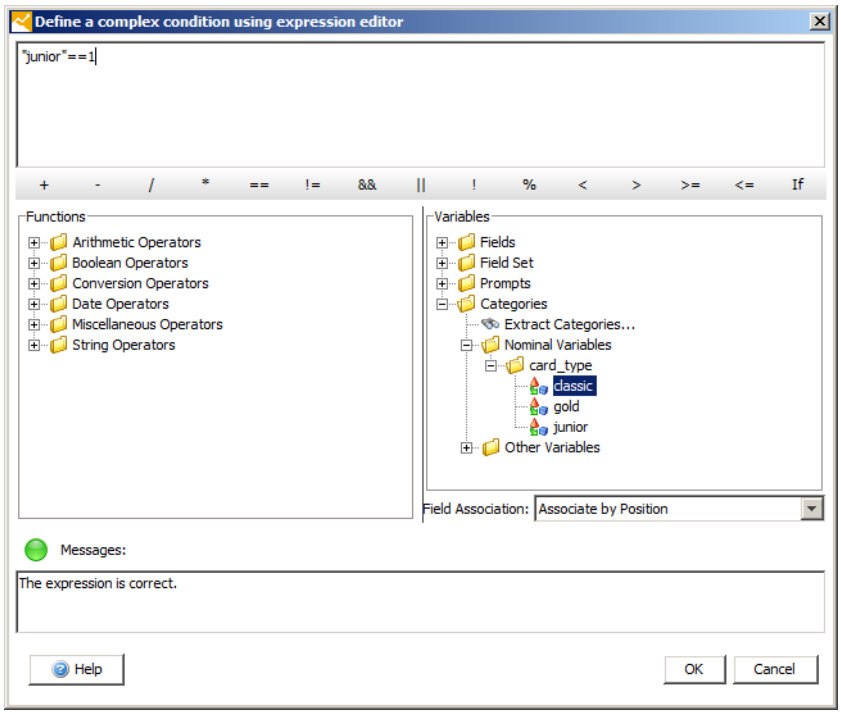
2 To add a new filter condition, click the button *New Condition*.



The expression editor opens.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

3 You can either use the offered *Functions* and *Variables* by double-clicking them, or you can enter directly a valid expression. The indicator above the *Messages* field indicates whether the expression is valid or not.



4 Click *OK* to create the condition. The panel listing the conditions is displayed.

5 Once all the conditions you need have been created, click *OK* to validate the filter.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

3.3.2 Creating the Analytical Record

Scenario

Advanced analytics best practice next step consists of creating an overall view of the entities (when this entity is a customer, it is sometimes called 'Customer Analytic Record', or '360 degree view of the customer', or even 'Customer DNA'). This view characterizes entities by a large number of attributes (the more, the better) which can be extracted from the database or even computed from events that occurred for each of them.

The list of all attributes corresponds to what is called an 'Analytical Record' of the entity 'disposition'. This analytical record can be decomposed into domains as listed below:

- Demographic
- Geo-demographic
- Complaints history
- Contacts history
- Product history
- Loan history
- Purchase history (coming from the transaction events)
- Segments
- Scores

.

Settings to Use

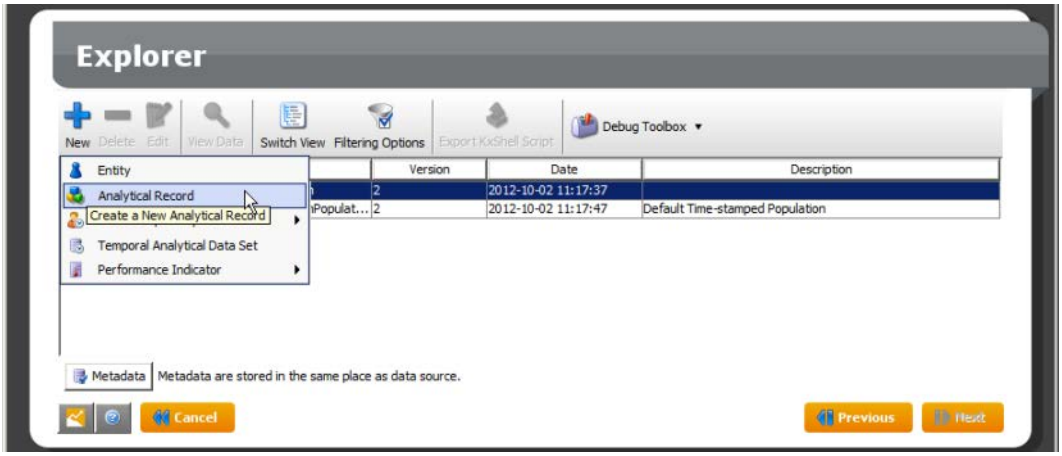
- Enter `AR_ClientGeoDemo` as the Analytical Record *Name*.
- Select `entity_disposition` in the list *Based on Entity*.
- Edit the following attributes in the *Merge* tab

Add information on...	Target table	Source Field	Target Field
the disposition	*disposition*	*kxId*	*disp_id*
the client	*client*	*client_id*	*client_id*
geographic and demographic data	*geodemo*	*home_geocode_id*	*geocode_id*
the credit card information	*credit_card*	*disp_id*	*disp_id*
the loan information	*loan*	*account_id*	*account_id*

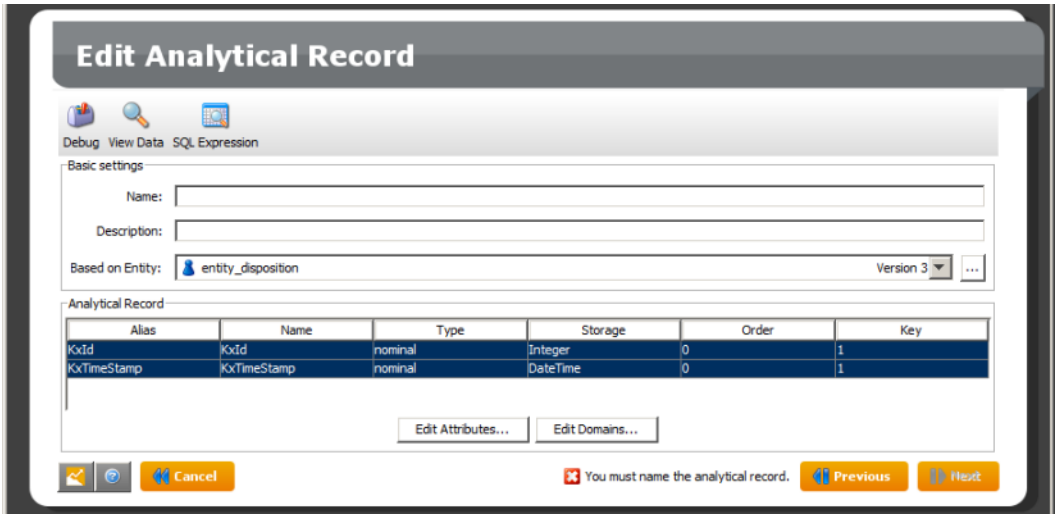
© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

☑ To Create an Analytical Record

1 In the panel *Explorer*, click the + menu and select *Analytical Record*.



The panel *Edit Analytical Record* is displayed.

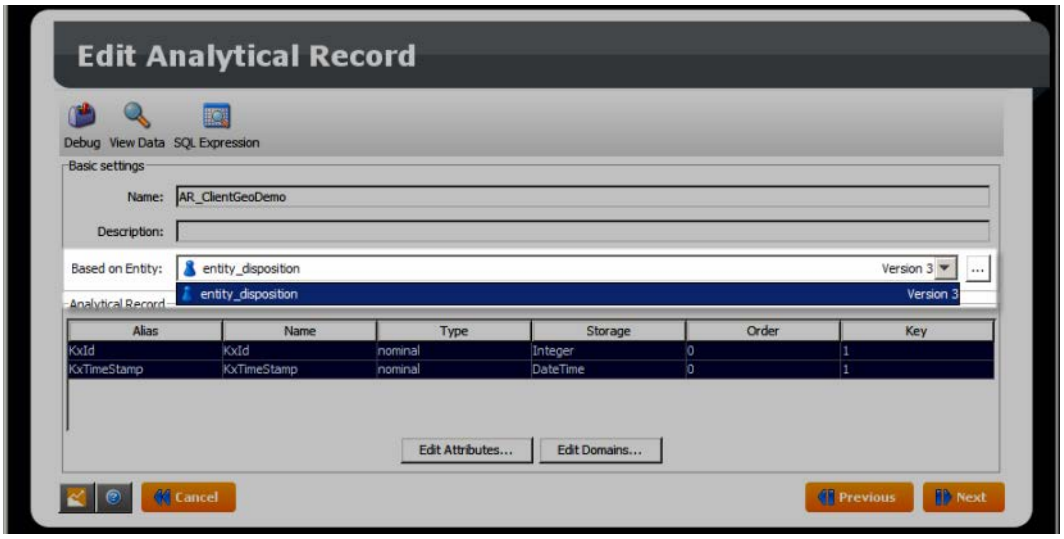


© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

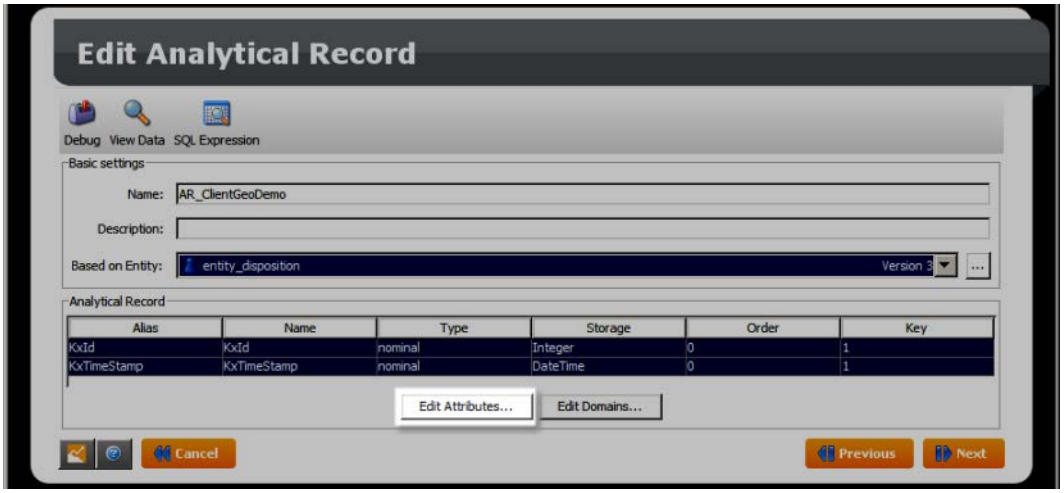
2 In the *Name* field, enter the name of the new analytical record.

3 In the *Description* field, enter a description for the new analytical record.

4 In the drop-down list *Based on Entity*, select the entity that will be used to create the new analytical record.



5 Click the button *Edit Attributes* to add computed fields or fields from other tables to the new analytical record.



For the detailed procedure, see **To Add Fields from Other Tables** below.

6 *Optional:* you can associate the analytical record attributes to specific domains. By default, the attributes are associated to a domain named after the table they are originated from.

7 Click the *Next* button to validate the creation or edition of the analytical record.

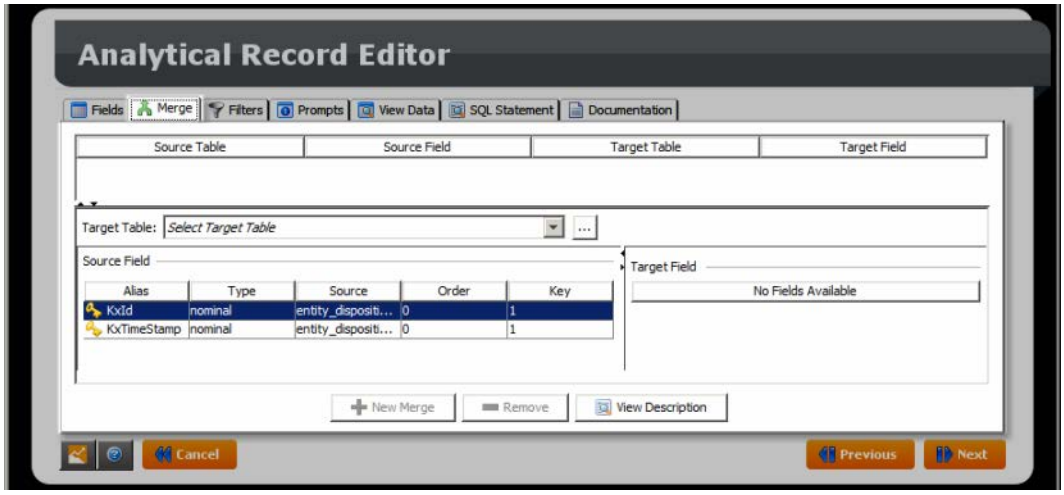
8 If an analytical record with the same name already exists, a dialog box is displayed asking you to choose what should be done:

- Click *Yes* to save the analytical record as a new version of the existing one,
- Click *No* to overwrite the existing analytical record.

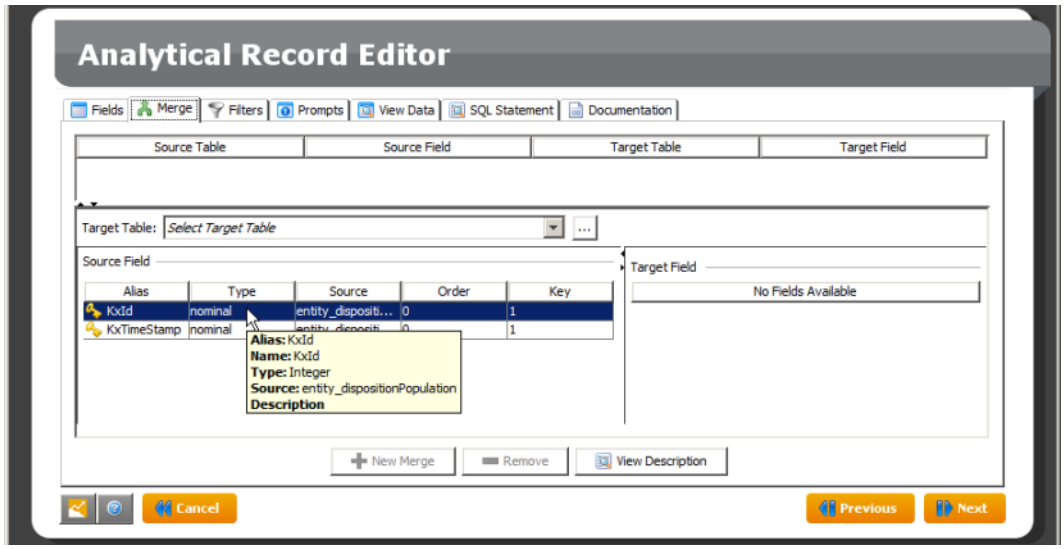
© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

☑ To Add Fields from Other Tables: Create a Merge

1 Select the *Merge* tab.

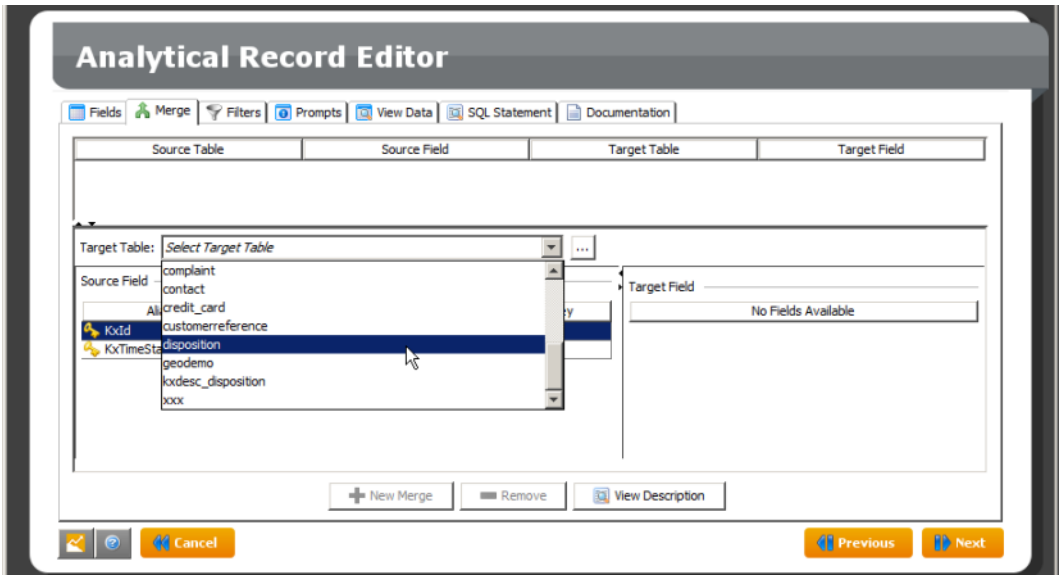


2 Select the source field in the list *Source Field*.

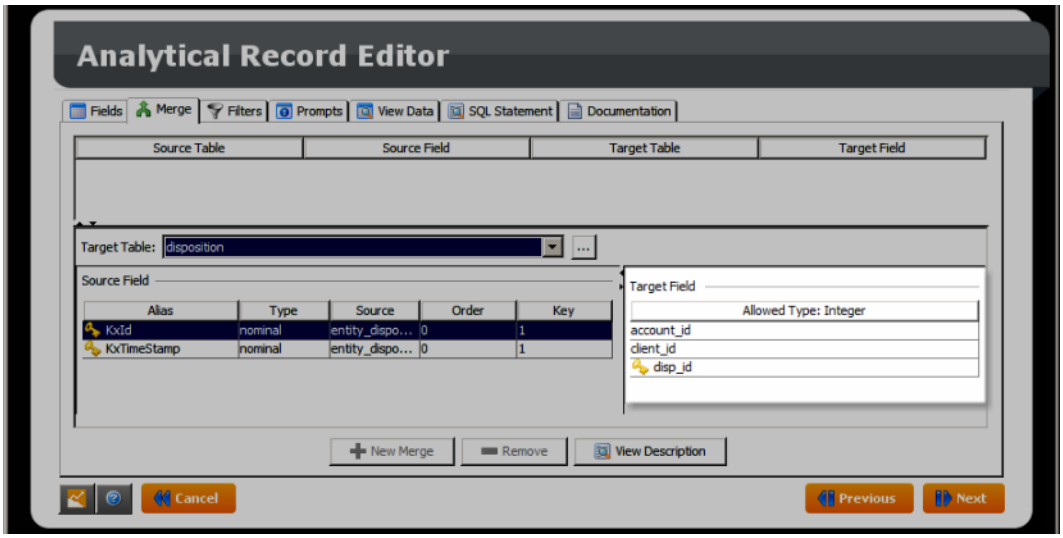


© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

3 Select the table to be joined in the list *Target Table*.

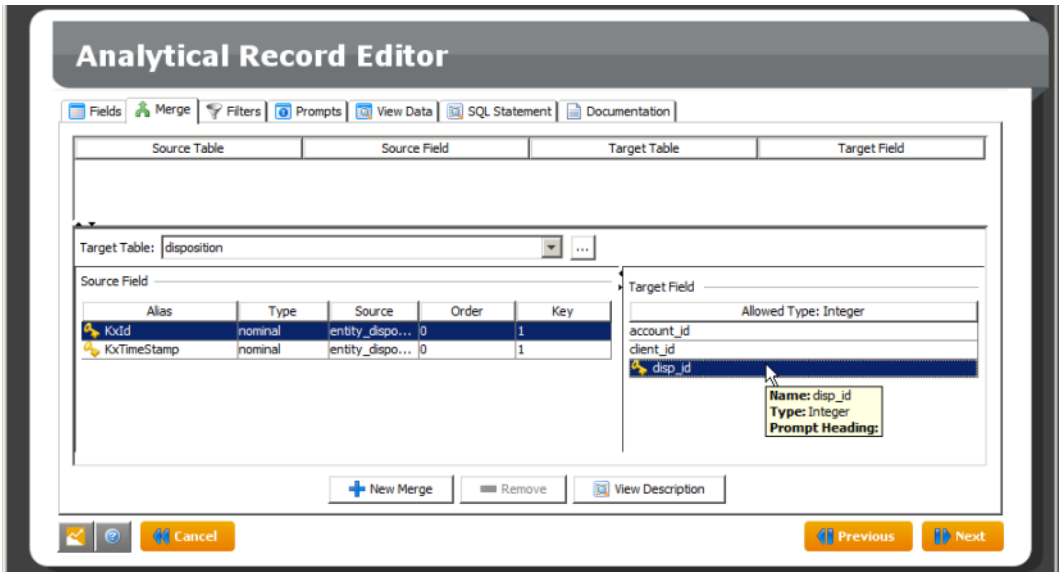


If the selected table contains fields corresponding to the source field type, they are displayed in the list *Target Field*, else the message *No Fields Available* is displayed.

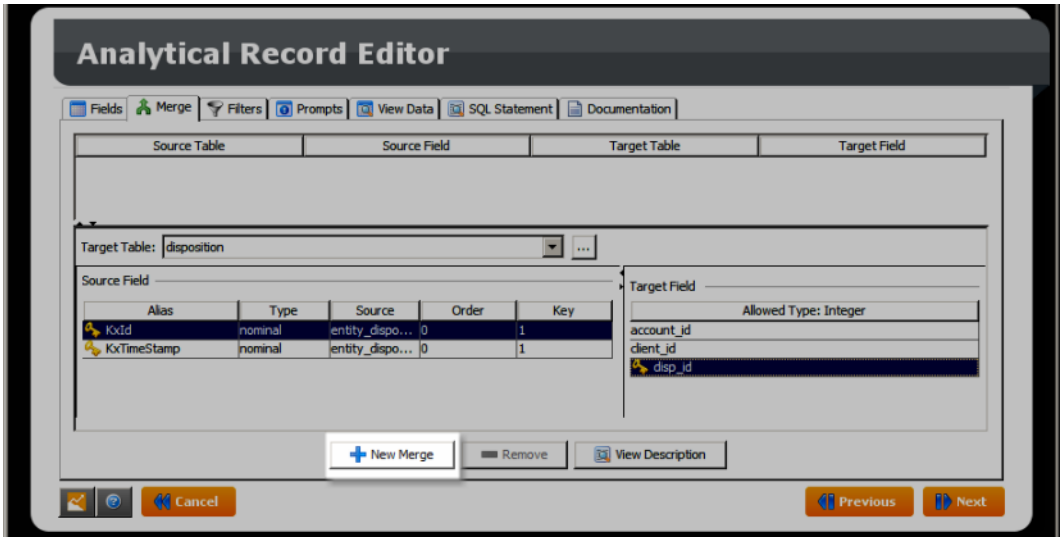


© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

4 Select the target field in the list.

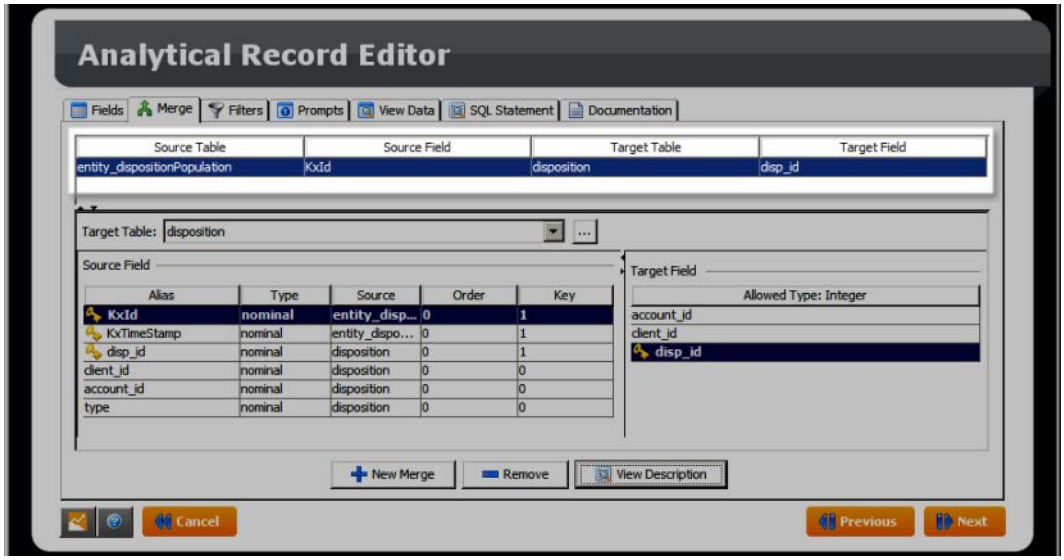


5 Click the button *New Merge* to create the merge.



© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

6 Repeat the procedure for each merge you want to create.
The existing merges are displayed in the upper part of the panel.



3.3.3 Updating the Default Time-stamped Population

Modifying the Default Time-stamped Population

Scenario

A default time-stamped population, named `entity_dispositionPopulation`, has been automatically created when you have created the `entity_disposition` entity. You will need to modify this population so that it contains all active dispositions for which no cards have been issued on January 1, 1998. An active disposition is a disposition for which at least one transaction has been performed in the 6 months prior to the selected date.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

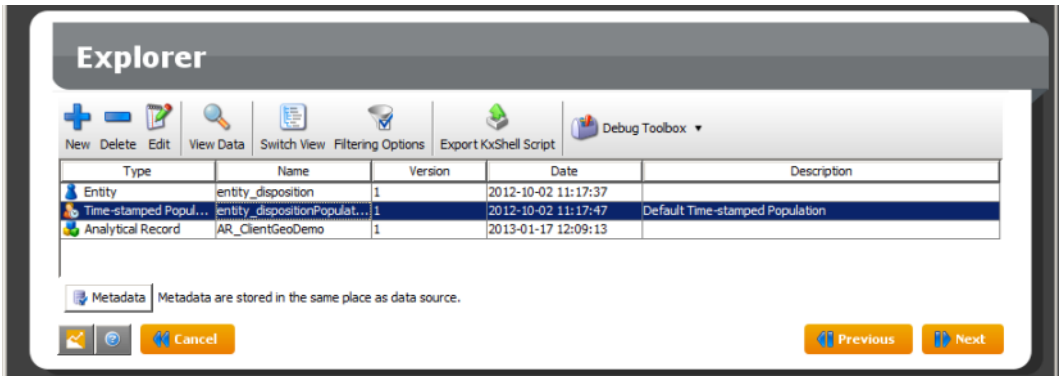
Editing a Time-stamped Population

ⓘ Settings to Use

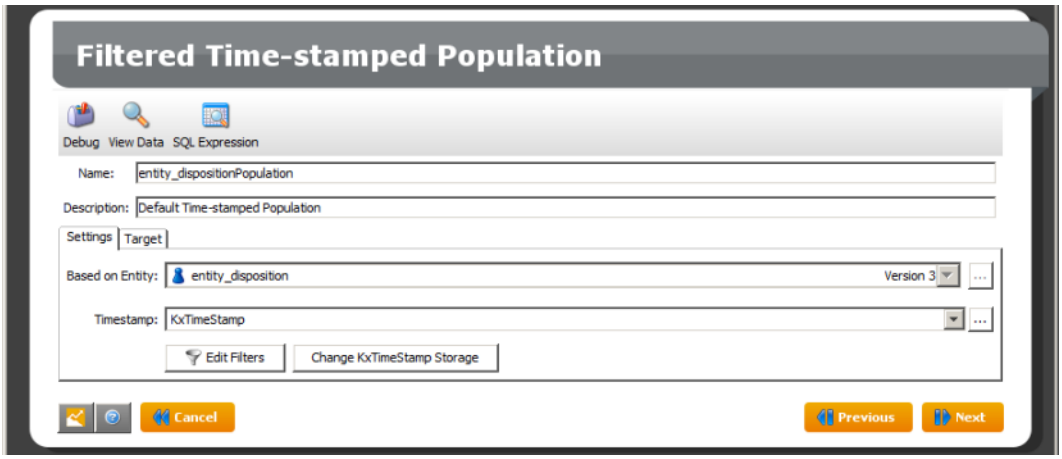
- *Name*: `ActiveDispositionNoCard@Date`
- *Based on Entity*: `entity_disposition`
- *Time Stamp*: `KxTimeStamp`

☑ To Edit a Time-stamped Population

1 In the *Explorer* panel, select the time-stamped population you want to edit.

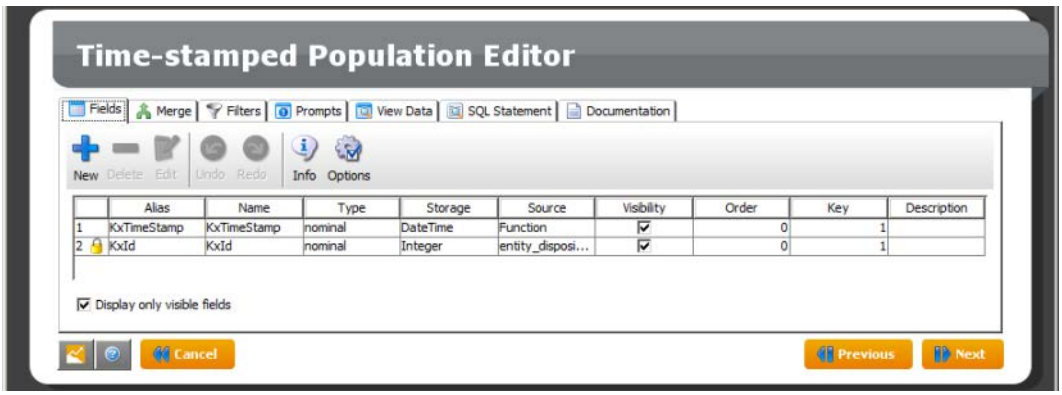


2 Click the button (*Edit*). The time-stamped population edition panel is displayed.



3 In the *Name* field, enter the name of the time-stamped population you want to create.

4 Click the button *Edit Filters*. The *Time-stamped Population Editor* opens.



© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

Editing the Time Stamp Prompt

Scenario

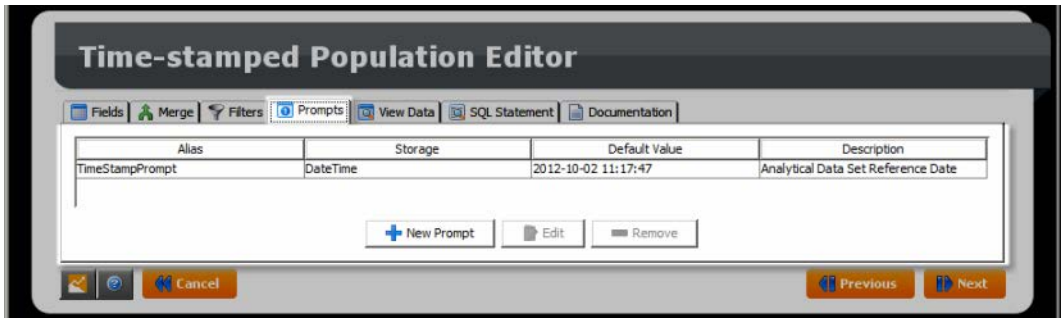
The time stamp created in the default time-stamped population is a prompt for which the default value is the date of its creation. You can choose to modify this prompt to better fit your needs.

Settings to Use

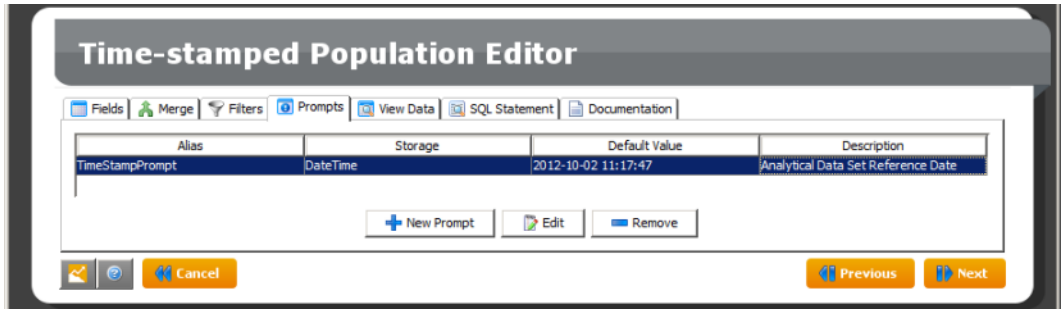
- *Default Value*: enter the date `1998-01-01 12:00:00`.
- *Prompt Heading*: do not change the default value.

To Edit the Time Stamp Prompt

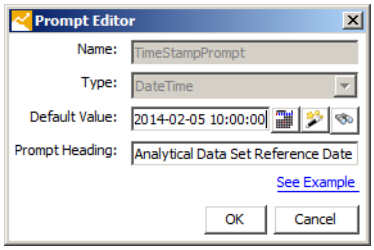
1 In the *Time-stamped Population Editor*, click the *Prompts* tab.



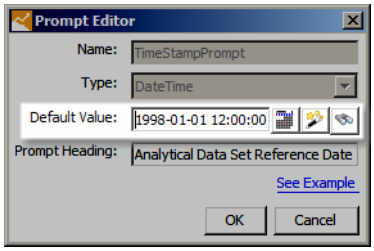
2 Select the prompt you want to edit.



3 Click the *Edit* button. The *Prompt Editor* opens.



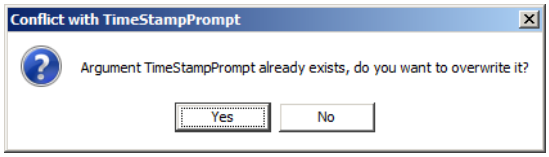
4 In the field *Default Value*, enter the value you want.



© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

5 In the field *Prompt Heading*, enter the text you want to see when the prompt is displayed.

6 Click *OK* to validate the changes. A dialog box is displayed asking you if you want to overwrite the existing prompt.



7 Click *Yes*. The changes are applied to the prompt.

Setting Filters on the Time-stamped Population

Creating a Merge

Scenario

To get the various identifiers linked to the dispositions (account, client) and the credit card information, you need to merge the entity used as basis for the time-stamped population with the table containing the dispositions and the table containing the credit card data.

Settings to Use

- The table containing the dispositions should already be merged with the time-stamped population
- For the merge with the credit cards table
  - *Target table*: `credit_card`
  - *Source Field*: `KxId`
  - *Target Field*: `disp_id`

To know how to create a merge, refer to the procedure **To Add Fields from Other Tables** (on page 25).

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

Creating an Aggregate

Scenario

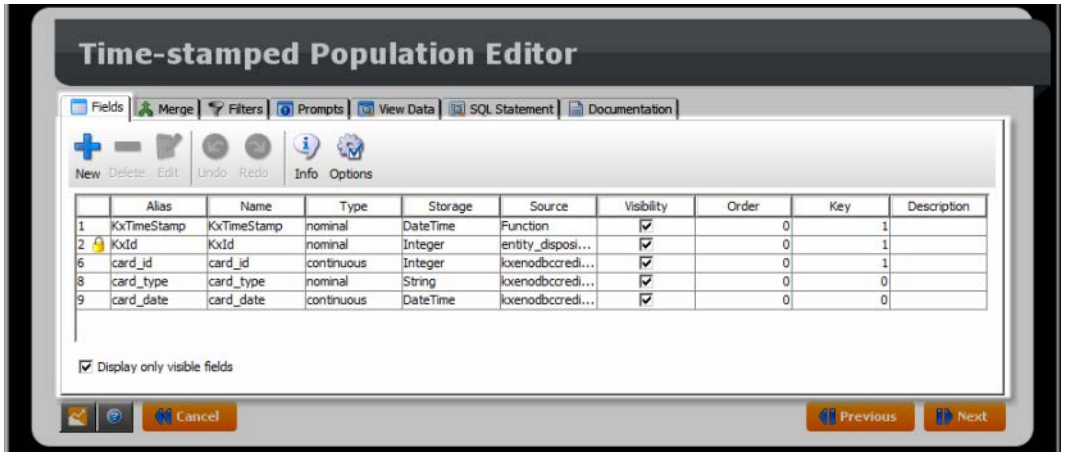
To select active dispositions, you need first to create an aggregate counting the number of transactions there has been for each disposition on the 6 months prior to the date of interest.

Settings to Use

Events Table	`transaction`
Events Date Column	`date_transaction`
Reference Table Key / Events Table Key	`account_id`
Function	`Count`
Target Column	`trans_id`
Period Settings	*Single Period*
Period Settings / Start Date	create a new field `KxTimeStampMinus6M` by using the Date Operator `Shift Date By Month` on the field `KxTimeStamp`, and shift back 6 months. You should get the following formula: `dateAddMonth(KxTimeStamp, -6)`.
Period Settings / End Date	`KxTimeStamp`
Name	`TransactionsNumberLast6M`

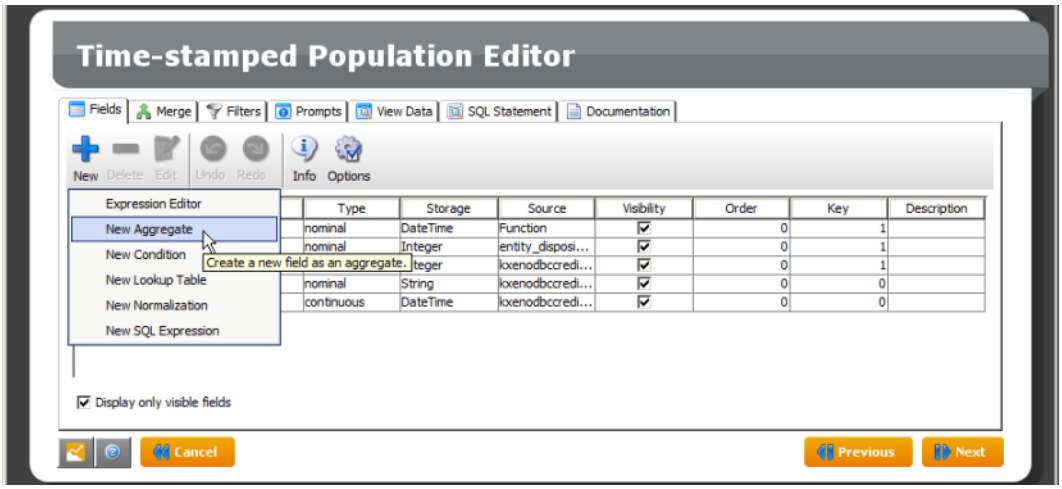
To Create a Field from an Aggregate

1 Select the *Fields* tab.



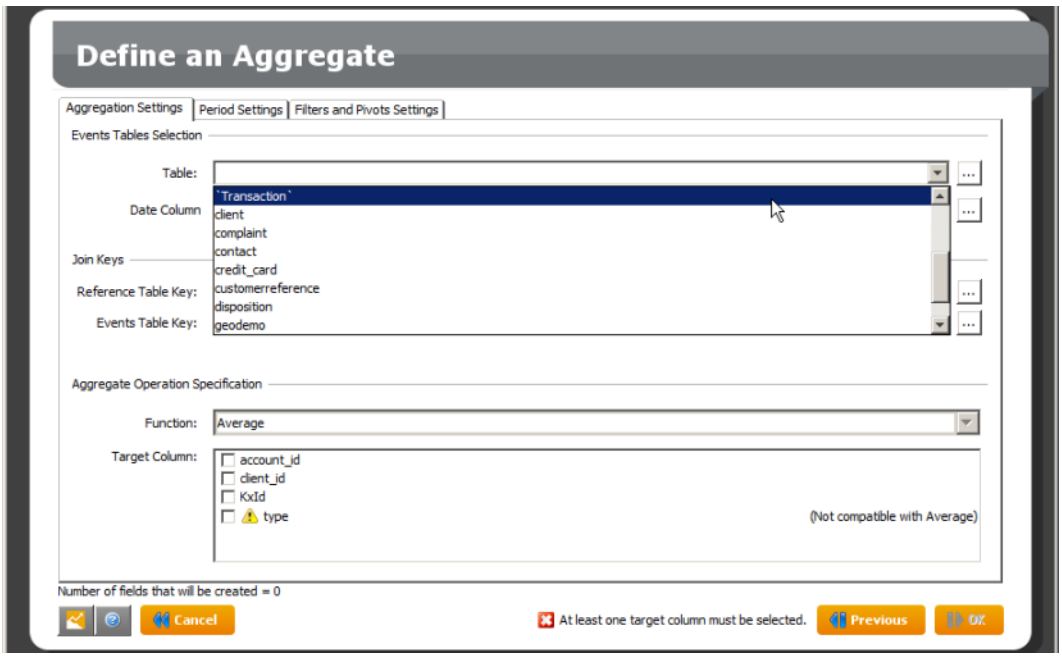
© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

2 In the + menu, select the option *New Aggregate*.



The panel *Define an Aggregate* is displayed.

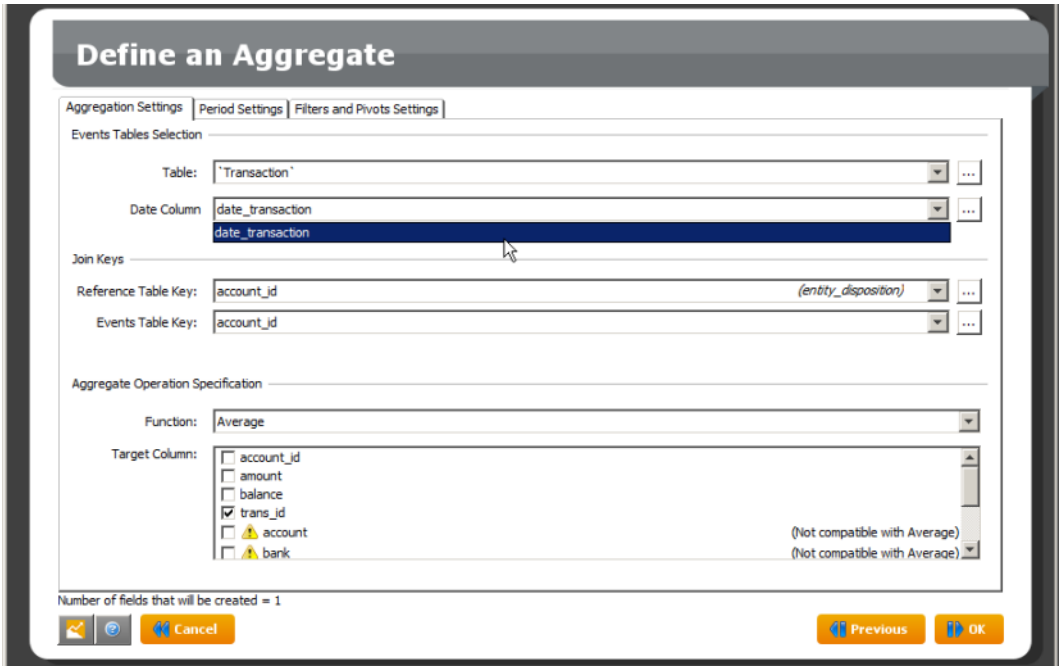
3 In the section *Event Tables Selection* of the tab *Aggregation Settings*, use the *Table* list to select the table that contains the event data you want to aggregate.



© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

4 If needed, enter an alias for the table in the field *Alias*.

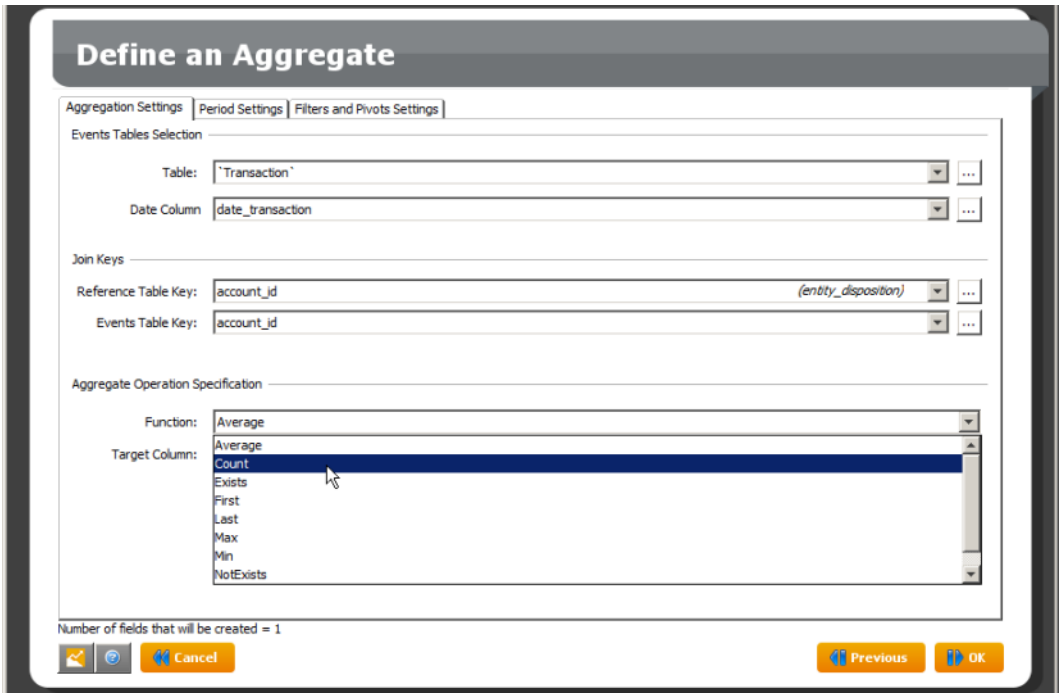
5 In the list *Date Column*, select the column that contains the event dates.



6 In the section *Join Keys*, select the reference table field to use for the join in the list *Reference Table Key*.

7 Select the events table field to use for the join in the list *Events Table Key*.

8 In the section *Aggregate Operation Specification*, use the *Function* list to select what kind of operation you want to do in the aggregate.

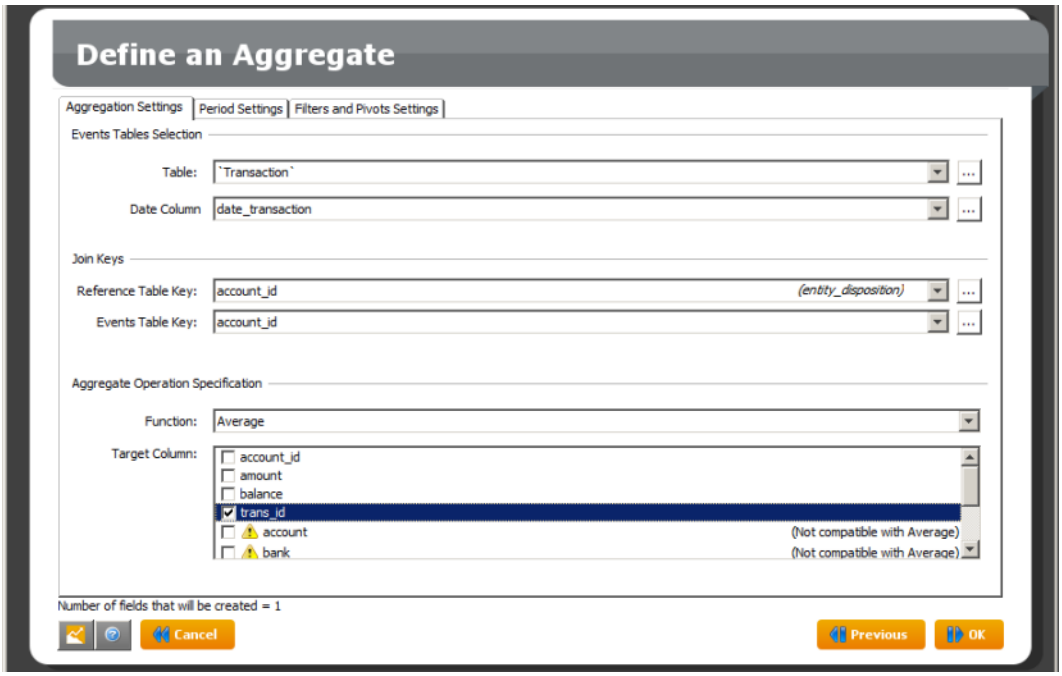


The following operations are available:

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

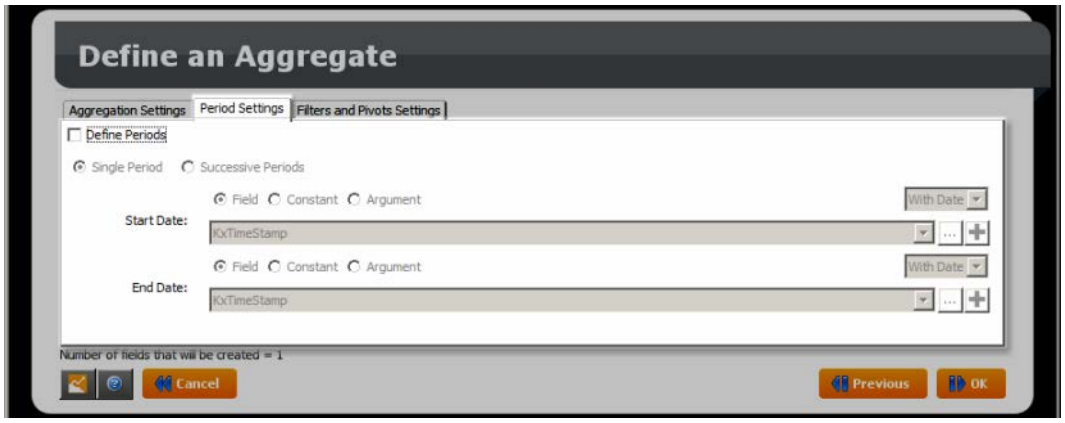
Functions	Description	Returned Values
Count	computes the number of occurrences	number of occurrences
Sum	computes the sum	sum
Average	computes the mean	mean
Min	identifies the minimum value	minimum value
Max	identifies the maximum value	maximum value
Exists	checks if at least one event exists for the current reference	0 if no event has been found 1 if at least one event has been found
NotExists	checks if no event exists for the current reference	0 if at least one event has been found 1 if no event has been found
First	identifies the first occurrence **Note** needs a date column	value of the first chronological occurrence for the current reference
Last	identifies the last occurrence **Note** needs a date column	value of the last chronological occurrence for the current reference

9 In the list *Target Column*, select which column to use for the aggregate.

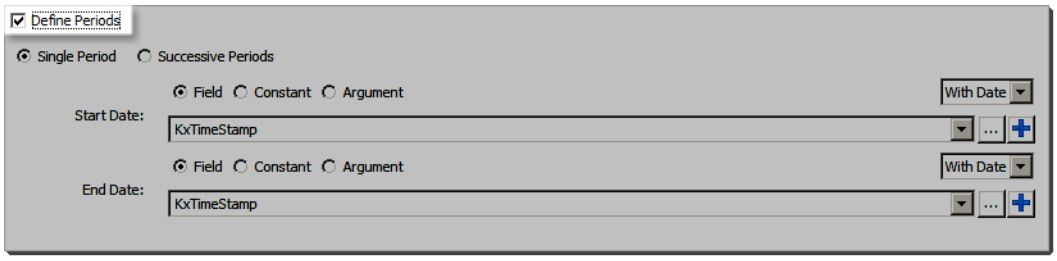


© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

10 If you want to define a time window for your aggregate, click the tab *Period Settings*.



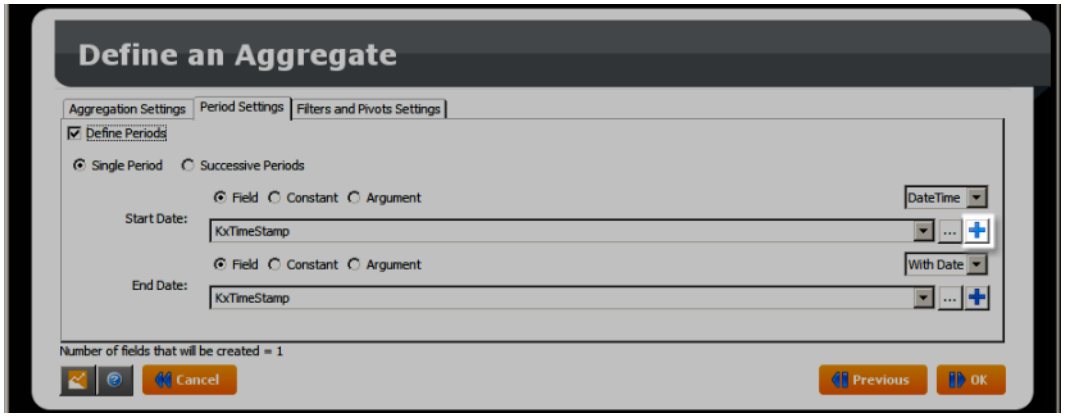
11 Check the box *Define Periods*.



12 Select if you want to define a *Single Period* or several *Successive Periods*.

- For a Single Period:
  Select a *Start Date* and an *End Date* for the period over which you want to aggregate the data. You can either use an existing field (*Field*), a constant (*Constant*) or a prompt (*Prompt*). If the field you want to use does not exist, you can create it on the fly by clicking the + button located on the right of the date field.

> **Note**
>
> The *start date is included* in the defined period, whereas the *end date is excluded* from it.



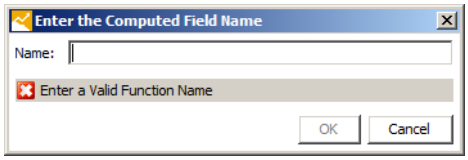
- For Successive Periods:
  Define the number of successive periods you want, their length and the starting date, using the hyperlinks (underlined in blue).

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

> **Note**
>
> The *starting date is included* in the periods.

13 Click *OK*.

14 Enter the name of the new field in the *Name* field.



15 Click *OK*.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

Filtering the Population on a Field

Scenario

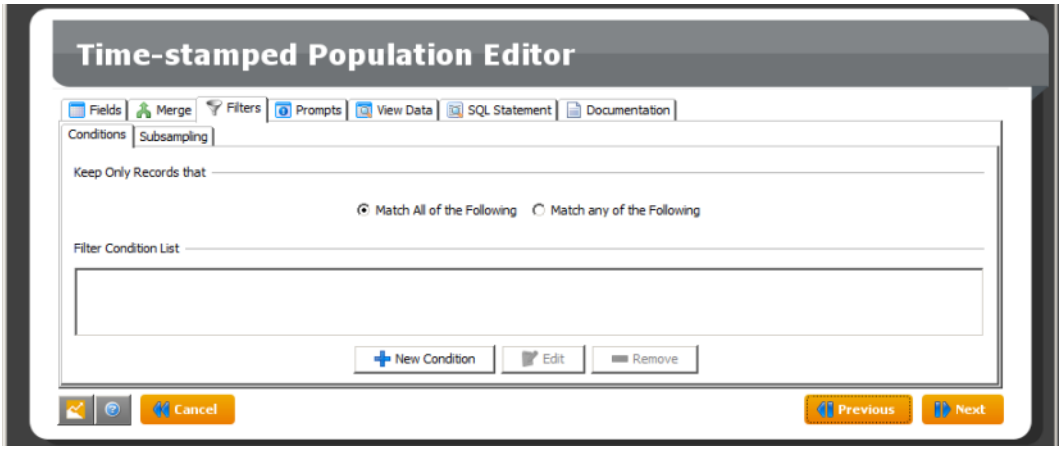
You will now filter the population to keep only the records of interest for your business issue. First, you need to filter the population to keep only the active dispositions, meaning that you will use the aggregate you just created to select only those having at least one transaction in the last 6 months. Then you will need to keep only the records for which there is no credit card on January 1, 1998; meaning the records for which the card issued date is either empty or is later than January 1, 1998.

Settings to Use

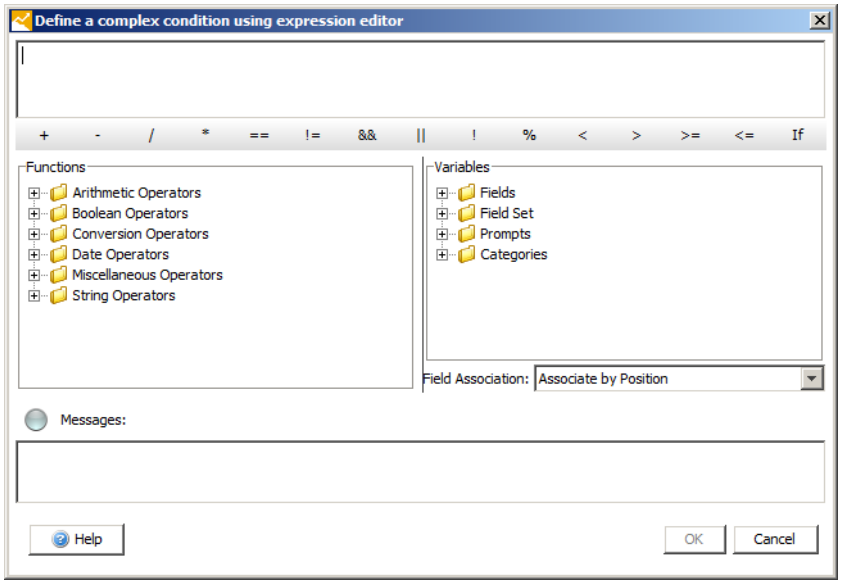
To Keep Only...	Create the Condition...
active dispositions	`TransactionsNumberLast6M > 0`
dispositions without credit cards	`isNull(issued_date) || issued_date > KxTimeStamp`

To Create a Condition

1 Click the button *New Condition*.

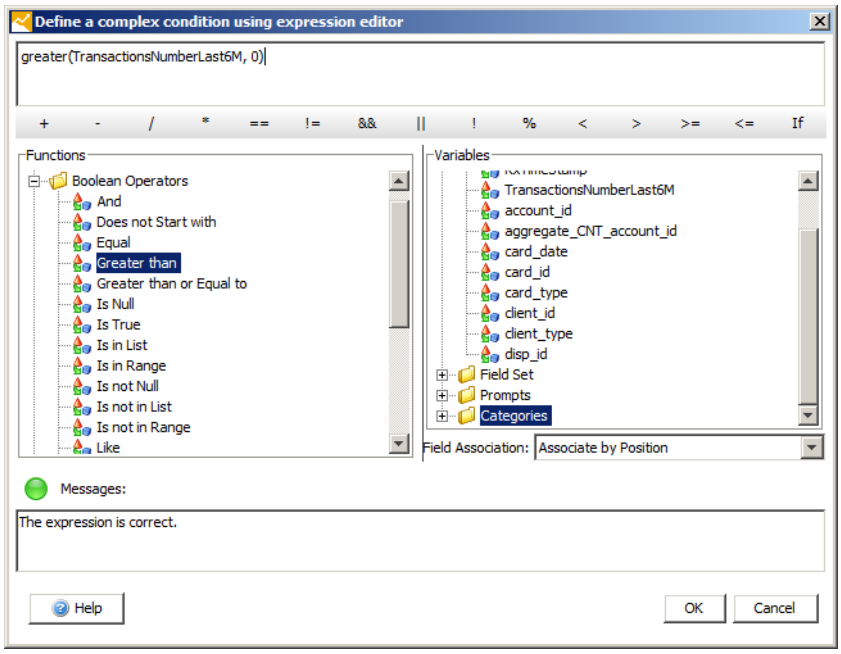


The expression editor opens.



© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

2 You can either use the offered *Functions* and *Variables* by double-clicking them, or you can enter directly a valid expression. The indicator above the *Messages* field indicates whether the expression is valid or not.



3 Click *OK*.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

Creating the Target

Scenario

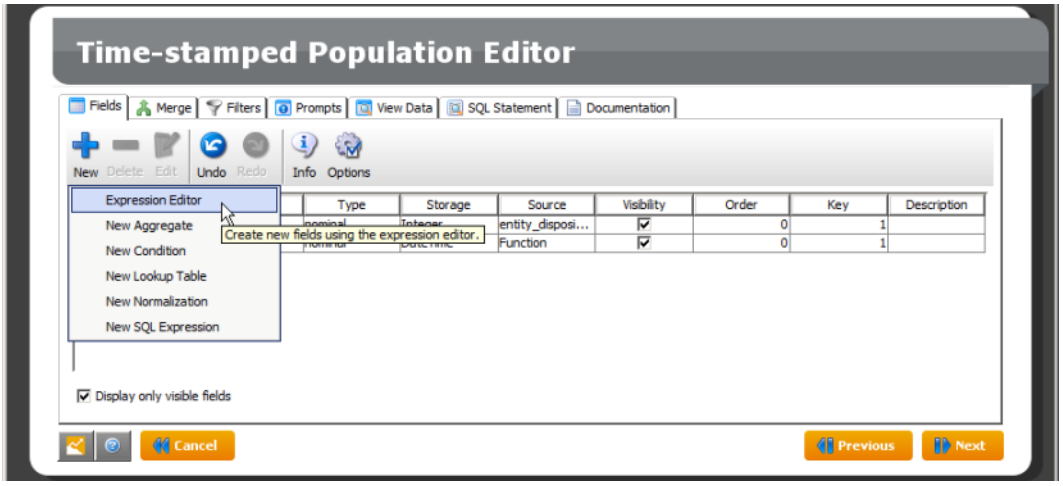
You want to create a model able to predict who is likely to buy a credit card in the following month. The time-stamped population you have created contains only dispositions that have no credit card associated on January 1, 1998, so you need to create a target indicating who has bought a credit card between January 1, 1998 and February 1, 1998.

Settings to Use

- In the *Expression Editor*, type the expression `KxTimeStamp<=issued_date && issued_date<dateAddMonth(KxTimeStamp, 1)`.
- *Computed Field Name*: enter `creditcard_bought`.

To Create the Variable to be Used as the Target

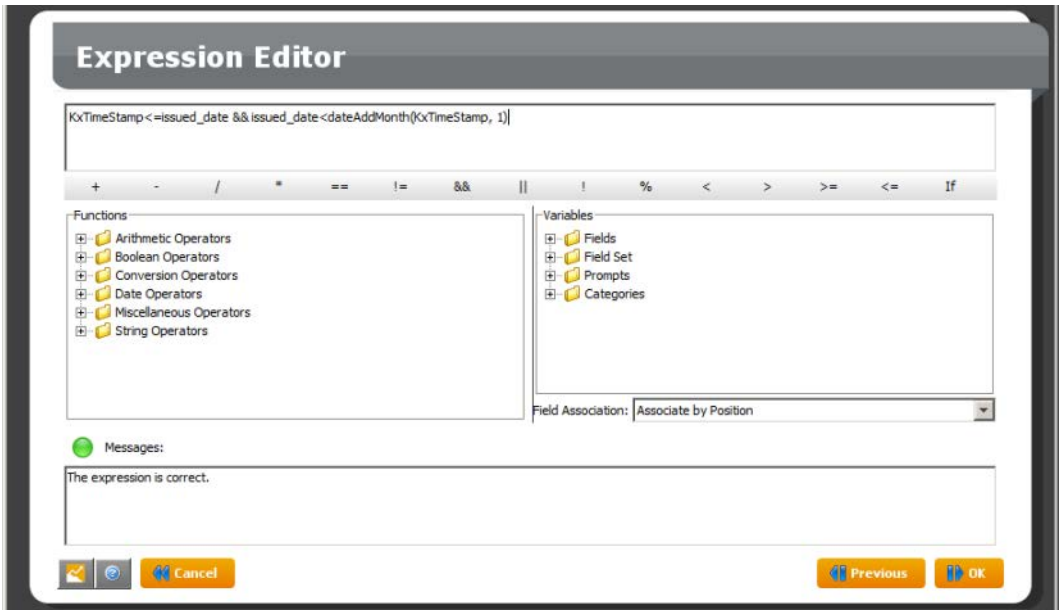
1. In the *Explorer* panel, select the time-stamped population to which you want to add a target.
2. Click the button (*Edit*). The *Filtered Time-stamped Population* panel is displayed.
3. Click the button *Edit Filters*. The *Time-stamped Population Editor* opens.
4. In the + menu, select the option *Expression Editor*.



The *Expression Editor* opens.

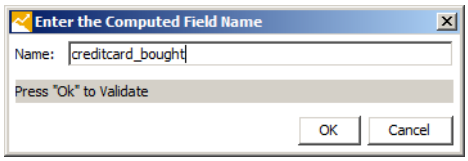
© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

5 You can either enter a valid expression directly in the upper text field, or you can insert functions, variables, field sets and prompts by double-clicking to insert them at the cursor location.



6 Click *OK*.

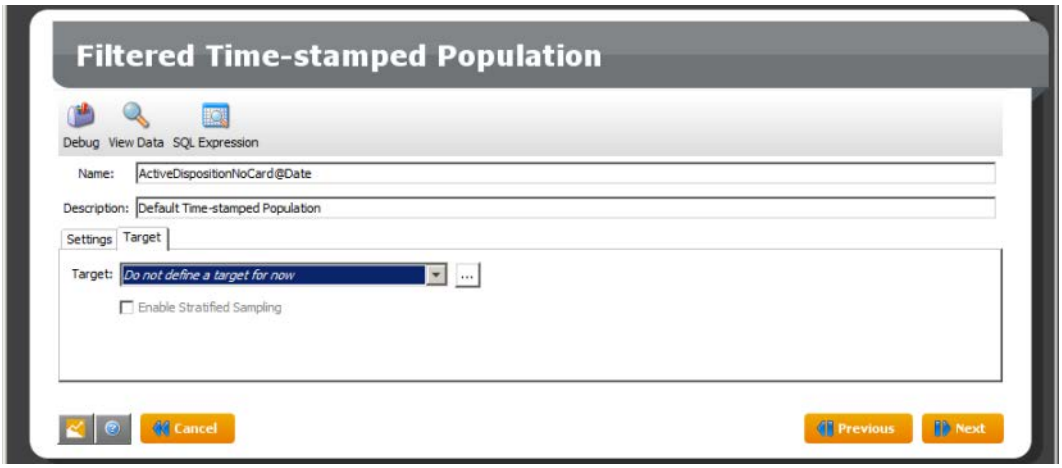
7 Enter the name of the variable in the *Name* field.



8 Click *OK* and then *Next* to define the variable.

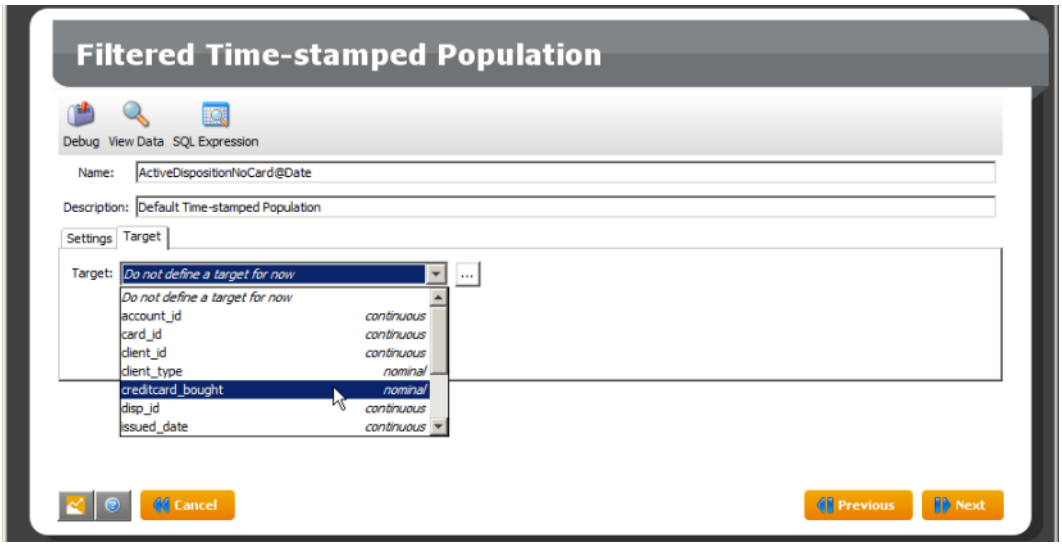
☑ To Define the Target

1 In the *Filtered Time-stamped Population* panel, click the *Target* tab.



© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

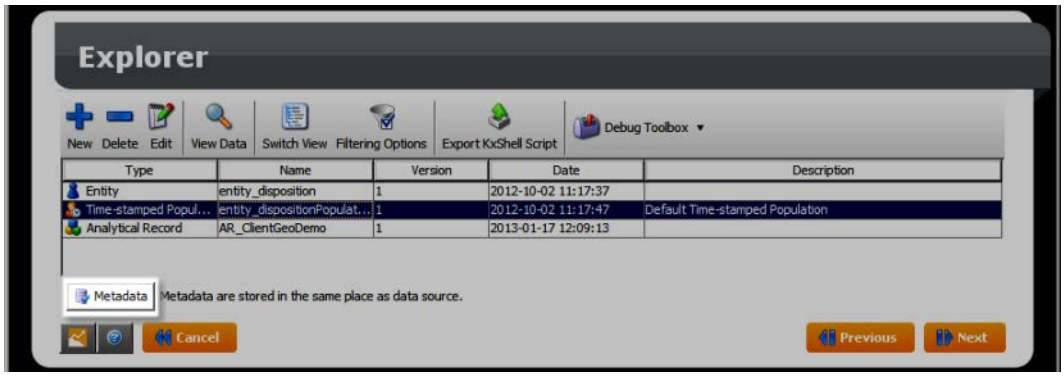
2 In the *Target* field, select the target variable you have just created.



3 Click *Next*.

3.3.4 Copying Objects to Another Metadata Repository

The button *Metadata* allows you to specify the location where the metadata should be stored. This Copy option can be useful when you are using different metadata depending on the team you are working with.

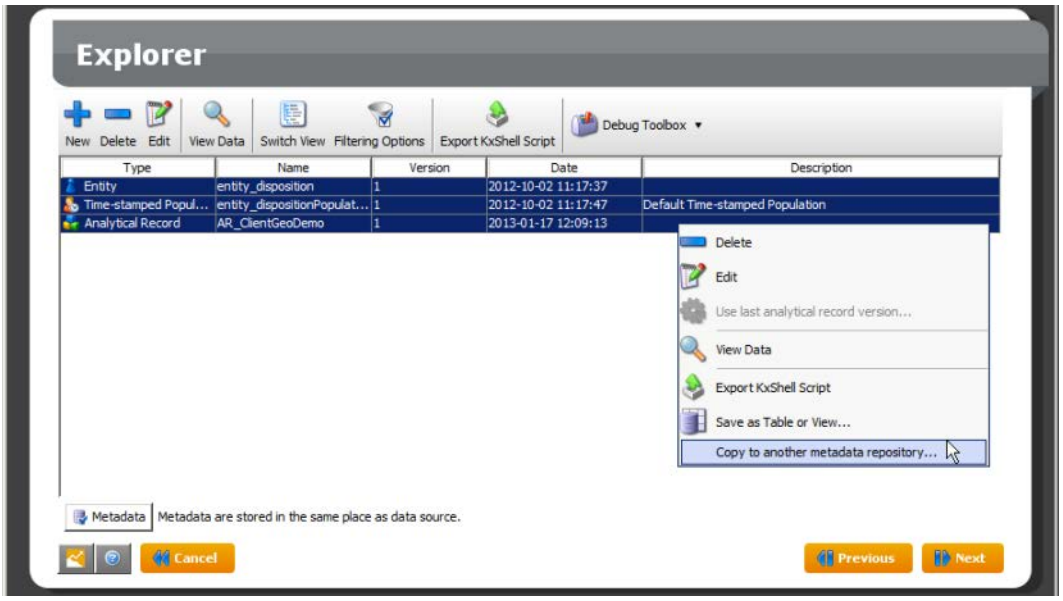


© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

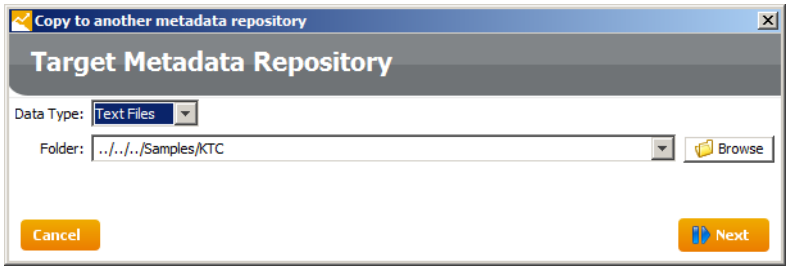
☑ To Copy Objects to Another Metadata Repository

1 Select the objects you want to copy.

2 Right-click on the selection and choose Copy to another metadata repository in the contextual menu.

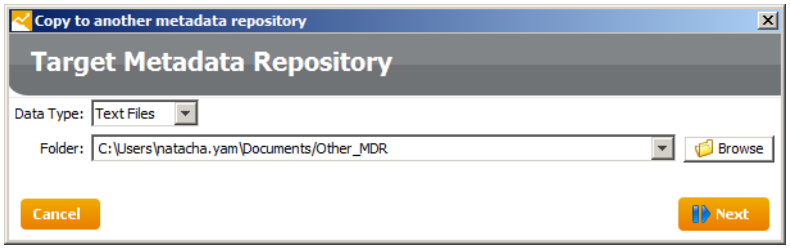


3 The window *Target Metadata Repository* is displayed.



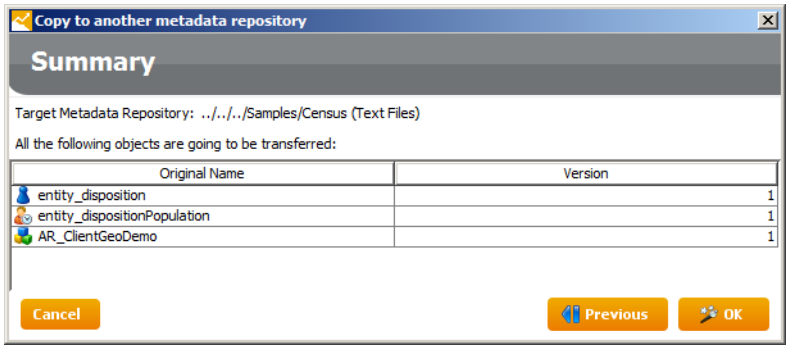
4 Select:

- a *Data Type*,
- a *Folder*

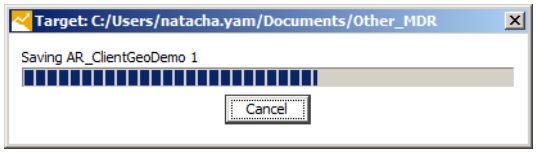


© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

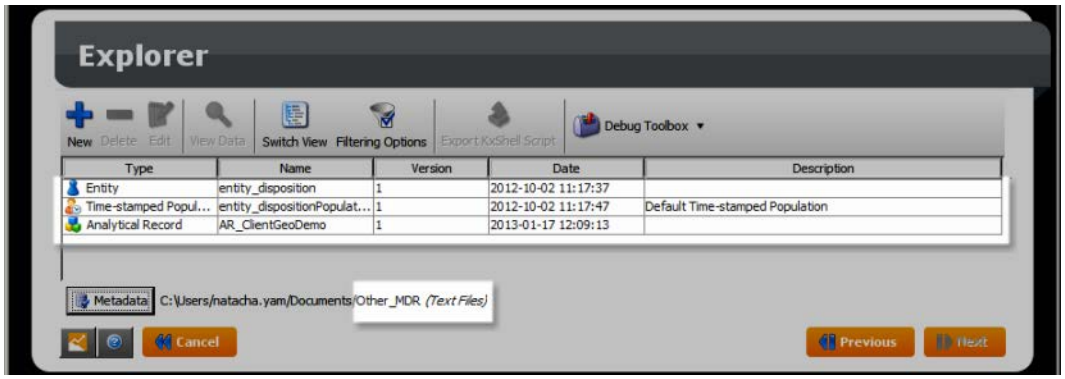
5 Click *Next*. A summary of the chosen Metadata repository and objects appears.



6 Click *OK* to confirm or Previous to change your former selection. If you confirm the summary, the objects will be saved in the new Metadata Repository.



7 To check that the procedure has been correctly carried out, select the target folder as the metadata repository in the *Explorer* panel. If the objects have been successfully saved, they will be displayed in the objects list.



3.4 Creating the First Classification Model

You will create a first classification model to see if the analytical record you have created contains sufficient information to predict who will be likely to buy a credit card in the following month.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

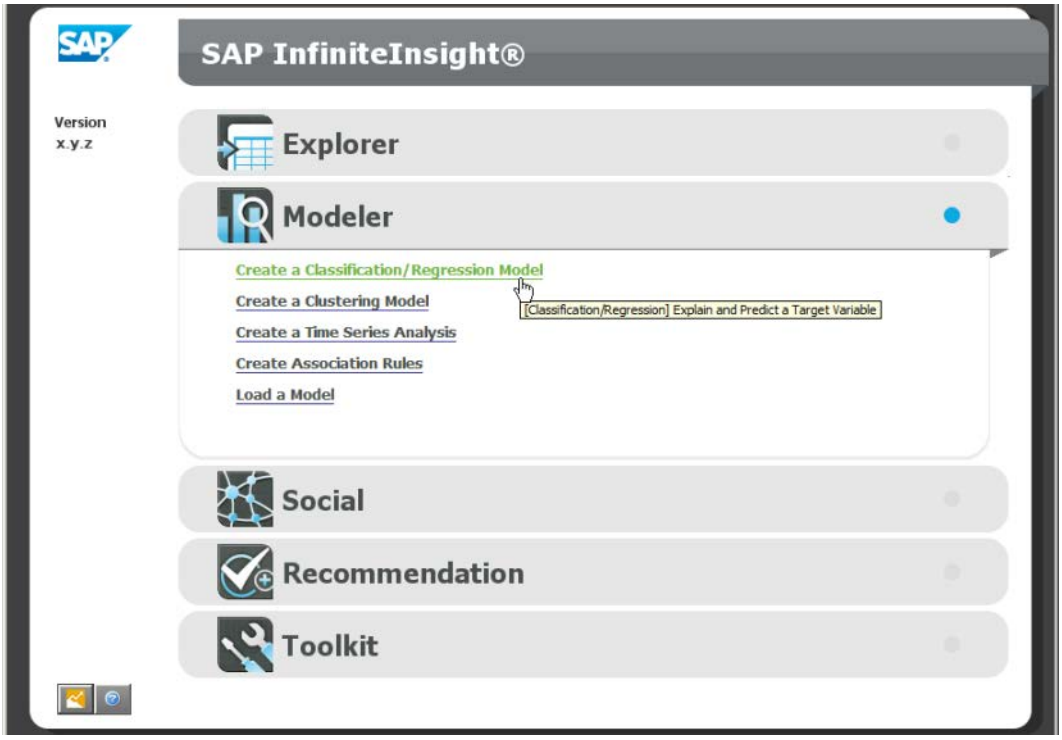
3.4.1 Selecting the Data to be Modeled

ⓘ Settings to Use

- Select the *Data Set Factory Mode*.
- *Database Source*: Select the database containing the data.
- *Analytical Record*: select the analytical record `AR_ClientGeoDemo`.
- *Time-stamped Population*: select the time-stamped population `ActiveDisposition@Date`.
- Validate the default date for the prompt.

☑ To Select the Data to be Modeled

1 In the *Modeler* section of InfiniteInsight® start panel, click the option *Create a Classification/Regression Model*.



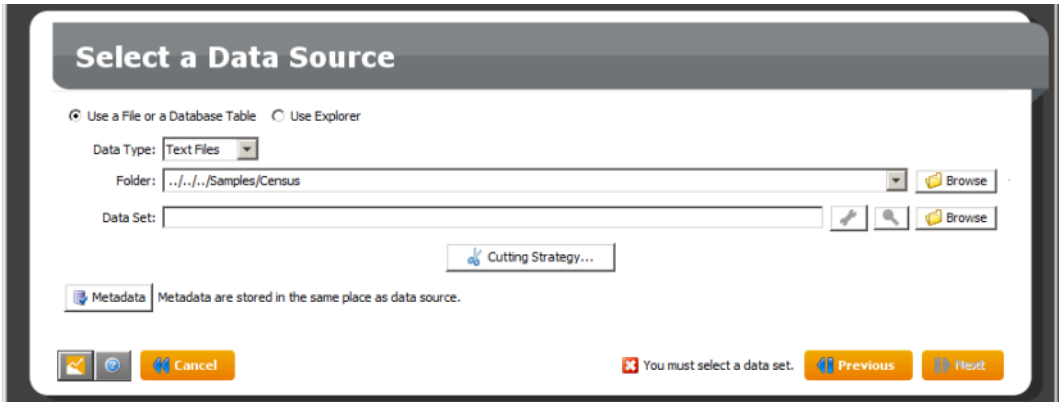
2 Check the mode you want to use to select the data to be modeled. There are two ways to select the data:

- *Use a File or Database Table*, which allows you to select a standard data source such as database table, a text file, or a data manipulation created with the SAP InfiniteInsight® Data Management feature.
- *Use Explorer*, which allows you to directly select the analytical record and the time-stamped population.

1. In *Use a File or Database Table*:

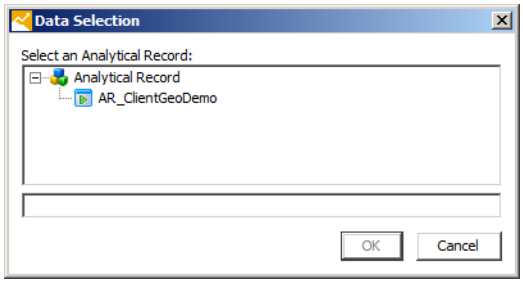
© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

- In the list *Data Type*, select the type of the data source.
- Use the *Browse* button corresponding to the *Folder* field to select the folder or database containing the data. If you want to access a protected database do not forget to enter the username and password when selecting the database.
- Use the *Browse* button corresponding to the *Data Set* field to select the file or table containing the data.

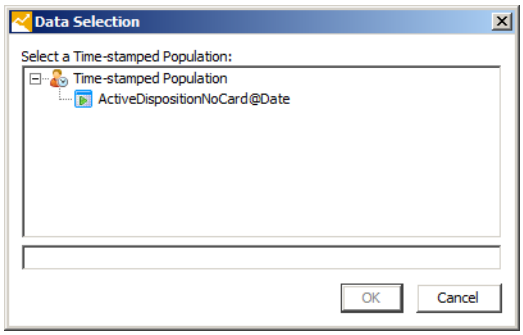


2. In *Use Explorer*:

- Use the *Browse* button corresponding to the field *Database Source* to select the database containing the analytical record and time-stamped population. If you want to access a protected database do not forget to enter the username and password when selecting the database.
- Use the *Browse* button corresponding to the field *Analytical Record* to select the analytical record you want to use. In the window *Data Source Selection*, all the existing Analytical Records are located under the tree item *Analytical Record*.



- Use the *Browse* button corresponding to the field *Time-stamped Population* to select the time-stamped population you want to use. In the window *Data Source Selection*, all the existing Time-stamped Populations are located under the tree item *Time-stamped Population*.

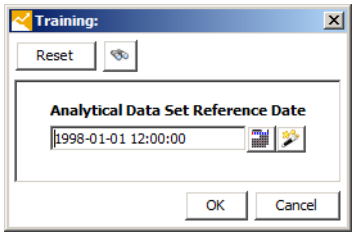


© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

3 If need be, create the target (see section Creating the Target on the Fly (see "Creating the Target" on page 40)).

4 Click the *Next* button. Depending on the size of the query sent to the database, an additional warning may be displayed, asking you to validate its execution. For more information see "Special Case: Data Stored in Databases - the Explain Mode" on page 47.

5 If the time stamp for the population is a prompt, it is displayed asking you to either validate the default date, or to enter another date.



6 Click the *OK* button.

Special Case: Data Stored in Databases - the Explain Mode

Before requesting data stored in a Teradata(1), Oracle(2) or SQLServer 2005 database, SAP InfiniteInsight® uses a feature, called the **Explain mode**, which categorizes the performances of SQL queries in several classes defined by the user. In order to be as fast and as light as possible, this categorization is done **without actually executing the full SQL query**.

Note

(1) For all versions of Teradata.
(2) For all versions above and including Oracle 10.

The objective is to allow you to estimate the workload of the SQL query before executing it and then deciding --possibly thanks to an IT Corporate Policy-- if the SQL query can actually be used.

For example, an IT Corporate Policy may favor interactivity and then define 3 classes of SQL queries, each with its maximum time:

- *Immediate*: *duration < 1 s*. The query is accepted and executed immediately.
- *Batched*: *1s <= duration < 2 s*. The query is accepted but will be executed on next idle time.
- *Rejected*: *2s <= duration*. The query will never be executed.

The number, names and limits of classes are defined by the user in order for these values to match the current DBMS configuration and DBMS usage policy.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

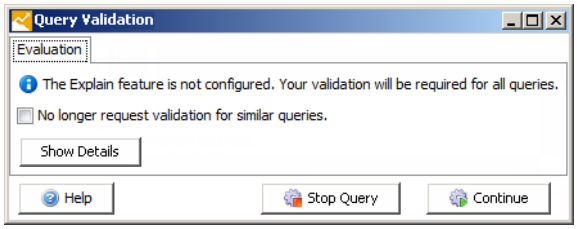
The Explain Mode has been Configured

If the Explain mode has been configured by your DBMS administrator, there are two possible outcomes to a query:

- **the query is accepted and executed**: this is completely transparent. SAP InfiniteInsight® accesses the data without further input from the user.
- **the query needs to be validated before being executed**: a pop-up window opens displaying a message configured by the DBMS administrator. A query that needs validation can be categorized in two ways:
  - **medium-sized**
    You will probably have to check with your administrator to know which action to take:
- If the administrator authorizes the query, click the *Continue* button. The pop-up window closes and the requested action is carried out.
- If the administrator does not authorize the query, click the button *Stop Query*, the pop-up window closes, but no action is executed.
  - **huge**
    It means that the query will take too much time and resources. In that case, the behavior of the *Continue* button depends on the configuration set by the DBMS Administrator (for example, it can automatically refuse queries that are considered too heavy). In any case, you should check with them to know the line of action to follow.

The Explain Mode has not been Configured

If your DBMS Administrator has not configured the Explain mode, the following pop-up opens when you try to access the data:



You need to contact your Administrator who will tell you which action to take and configure the Explain mode.

If the Administrator validates the execution of the query, you may want all queries with the same duration to be executed without validation. In that case, check the box *Do not request validation anymore for similar requests*. The validation message will then only appear for larger queries. This configuration will only be used for the current session, when closing SAP InfiniteInsight®, it will be lost. For a permanent configuration, see your DBMS Administrator, who will find the necessary information in the support document Explain Mode available in section Support and Integration Documentation of SAP InfiniteInsight® documentation.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

3.4.2 Describing the Data

In order for SAP InfiniteInsight® components to interpret and analyze your data, the data must be described. To put it another way, the description file must specify the nature of each variable, determining their:

- **Storage** format: number (*number*), character string (*string*), date and time (*datetime*) or date (*date*).
- **Type:** *continuous*, *nominal* or *ordinal*.

For more information about data description, see **Types of Variables** and **Storage Formats** in the **SAP InfiniteInsight® User Guide**.

To describe your data, you can:

- Either **use an existing description file**, that is, taken from your information system or saved from a previous use of SAP InfiniteInsight® components,
- Or **create a description file using the** *Analyze* **option**, available to you in SAP InfiniteInsight® Modeling Assistant. In this case, it is important that you validate the description file obtained. You can save this file for later reuse. If you name the description file **KxDoc_<SourceFileName>**, it will be automatically loaded when clicking the *Analyze* button.

Caution

The description file obtained using the *Analyze* option results from the analysis of the first 100 lines of the initial data file. In order to avoid all bias, we encourage you to mix up your data set before performing this analysis.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

Each variable is described by the fields detailed in the following table:

The Field...	Gives information on...
Name	the variable name (which cannot be modified)
Storage	the type of values stored in this variable: ▪ *Numbe*r: the variable contains only "computable" numbers (be careful a telephone number, or an account number should not be considered numbers) ▪ *String*: the variable contains character strings ▪ *Datetime*: the variable contains date and time stamps ▪ *Date*: the variable contains dates
Value	the value type of the variable: ▪ *Continuous*: a numeric variable from which mean, variance, etc. can be computed ▪ *Nominal*: categorical variable which is the only possible value for a string ▪ *Ordinal*: discrete numeric variable where the relative order is important
Key	whether this variable is the key variable or identifier for the record: ▪ *0* the variable is not an identifier; ▪ *1* primary identifier; ▪ *2* secondary identifier...
Order	whether this variable represents a natural order. There must be at least one variable set as Order in the Event data source. Warning - If the data source is a file and the variable stated as a natural order is not actually ordered, an error message will be displayed before model checking or model generation.
Missing	the string used in the data description file to represent missing values (e.g. "999" or "#Empty" - without the quotes)
Group	the name of the group to which the variable belongs
Description	an additional description label for the variable

ⓘ Settings to Use

- Use the *Analyze* feature.
- Set all the identifiers to *nominal*.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

3.4.3 Selecting the Variables

Scenario

By default, and with the exception of key variables (such as *KxIndex*), all variables contained in your data set are taken into consideration for generation of the model. You may exclude some of these variables.

For the first analysis of your data set, we recommend that you **retain all variables**. It is particularly important to retain even the variables that seem to have no impact on the target variable. If indeed these variables have no impact on the target variable, the model will confirm this. In the contrary case, the model will allow you to recognize previously unidentified correlations between these variables and the target variable. By excluding variables from the analysis based on simple intuition, you take the risk of depriving yourself of one of the greatest value-added features of SAP InfiniteInsight® models: the discovery of **non-intuitive information**.

To create your model, you use data where the target is already known, however to be able to predict correctly your target when you apply the model, you need to exclude from the explanatory variables any information that allowed you to create the target. In this scenario, it means that you need to remove from the explanatory variables all the variables coming from the table in which the credit card information is stored.

Settings to Use

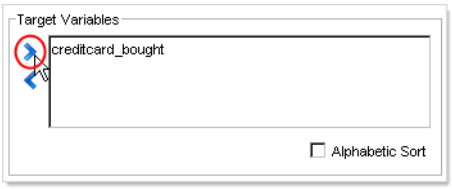
- *Target Variable*: `creditcard_bought`
- *Weight Variable*: none
- *Excluded Variables*: `KxIndex_1`, `card_id`, `card_type`, `issued_date`

To Select the Target Variable

1 On the screen *Selecting Variables*, in the section *Explanatory Variables Selected* (left hand side), select the variables you want to use as Target Variables. By default, the last variable of the data set is selected as the target variable.

> **Note**
>
> On the screen *Selecting Variables*, variables are presented in the same order as that in which they appear in the table of data. To sort them alphabetically, select the option *Alphabetic sort*, presented beneath each of the variables list.

2 Click the button > located on the left of the screen section *Target(s) Variable(s)* (upper right hand side).



The variable moves to the screen section *Target(s) Variable(s)*.
Also, select a variable in the screen section *Target(s) Variable(s)* and click the button < to move the variables back to the screen section *Explanatory Variables Selected*.

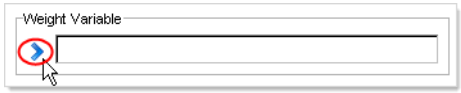
To Select a Weight Variable

3 On the screen *Selecting Variables*, in the section *Explanatory Variables Selected* (left hand side), select the variables you want to use as a Weight Variable.

> **Note**
>
> On the screen *Selecting Variables*, variables are presented in the same order as that in which they appear in the table of data. To sort them alphabetically, select the option *Alphabetic sort*, presented beneath each of the variables list.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

4 Click the button > located on the left of the screen section *Weight Variable* (middle right hand side).



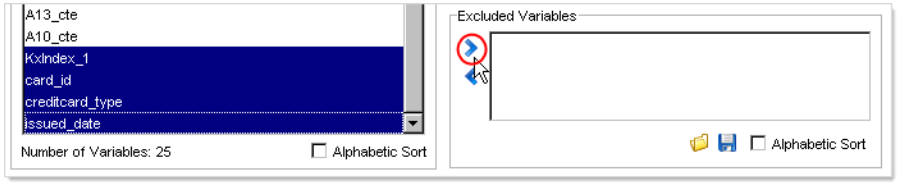
The variable moves to the screen section *Weight Variable*.
Also, select a variable in the screen section *Weight Variable* and click the button < to move the variables back to the screen section *Explanatory Variables Selected*.

☑ To Exclude Variables

1 On the screen *Selecting Variables*, in the section *Explanatory variables selected* (left hand side), select the variable to be excluded.

> **Note**
>
> On the screen *Selecting Variables*, variables are presented in the same order as that in which they appear in the table of data. To sort them alphabetically, select the option *Alphabetic sort*, presented beneath each of the variables list.

2 Click the button > located on the left of the screen section *Excluded Variables* (lower right hand side).



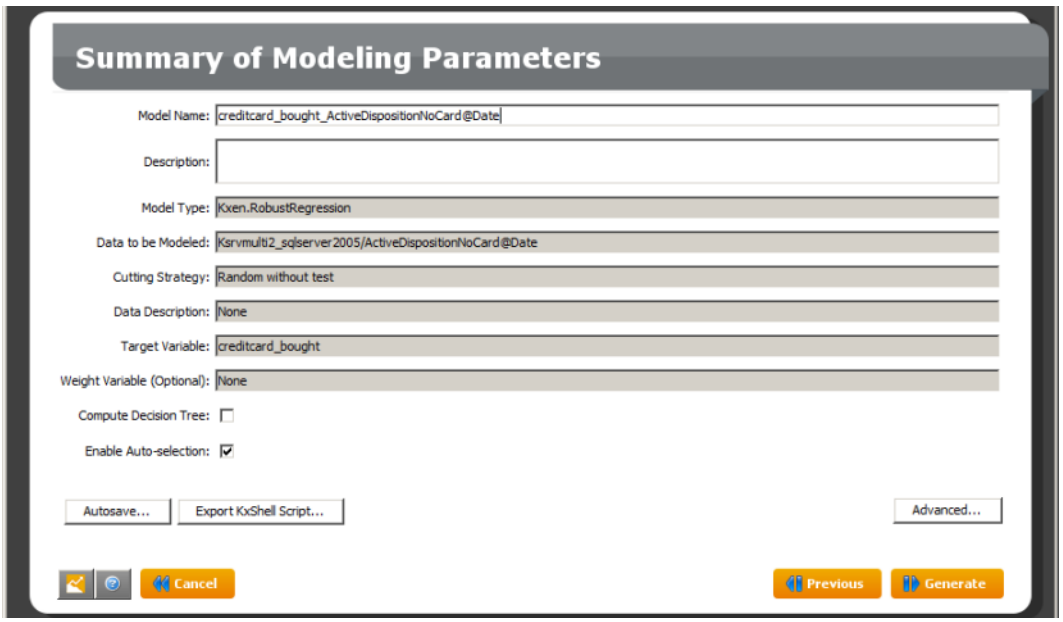
The variable moves to the screen section *Variables excluded*.
Also, select a variable in the screen section *Variables excluded* and click the button < to move the variables back to the screen section *Explanatory variables selected*.

3 Click the *Next* button.
The screen *Parameters of the Model* appears.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

3.4.4 Checking the Model Parameters

The screen *Summary of Modeling Parameters* allows you to check the modeling parameters just before generating the model.



The name of the model is automatically filled. It corresponds to the name of the target variable, followed by the underscore sign ("_") and the name of the data source, minus its file extension.

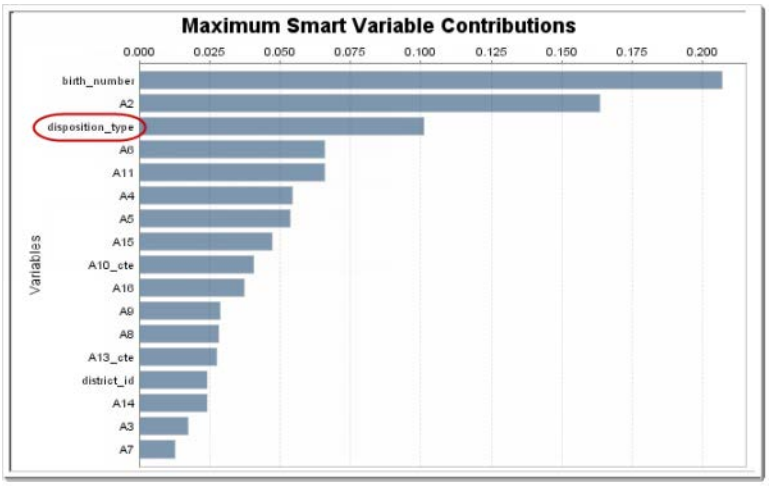
> **Note**
>
> The screen *Summary of Modeling Parameters* contains an *Advanced* button. By clicking this button, you access the screen *Specific Parameters of the Model*, where you can set the polynomial order of the model to be generated. For more information, see the K2R section of the **SAP InfiniteInsight® User Guide**.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

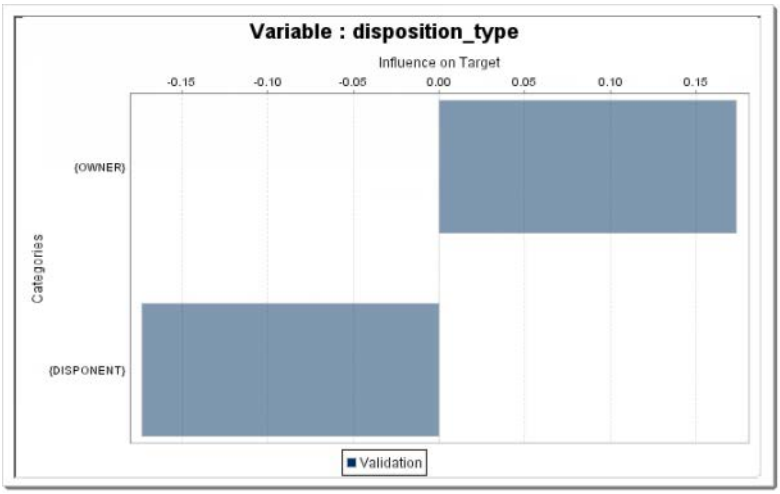
3.5 Analyzing the Model Results

If you look at your model results, you can see that the performance indicators are not very high. The quality of your model, represented by the KI, can be improved by adding information to the analytical record. The analytical record you have created contained only geographic and demographic data, however you have a lot of information available on the customers behavior, such as the amount withdrawn from an account or the amount checked in the account, the balance at the end of each month, and the tenure of the account. You could also add the product history, with the number of loans taken in the last year for example, and the loan history.

However, this first model already reveals some information. If you look at the *Contributions by Variables* (see graphic below), you can see that one of the most important variables is the disposition type.



If you click this variable on the graph, you will display the categories significance, that will show that a customer who is owner of an account is more likely to buy a credit card, than one who is only a user of the account.



© 2014 SAP SE or an SAP affiliate company. All rights reserved-Application Scenario: Segmented Cross-sell in Retail Banking

4 Performing an Entity Transfer

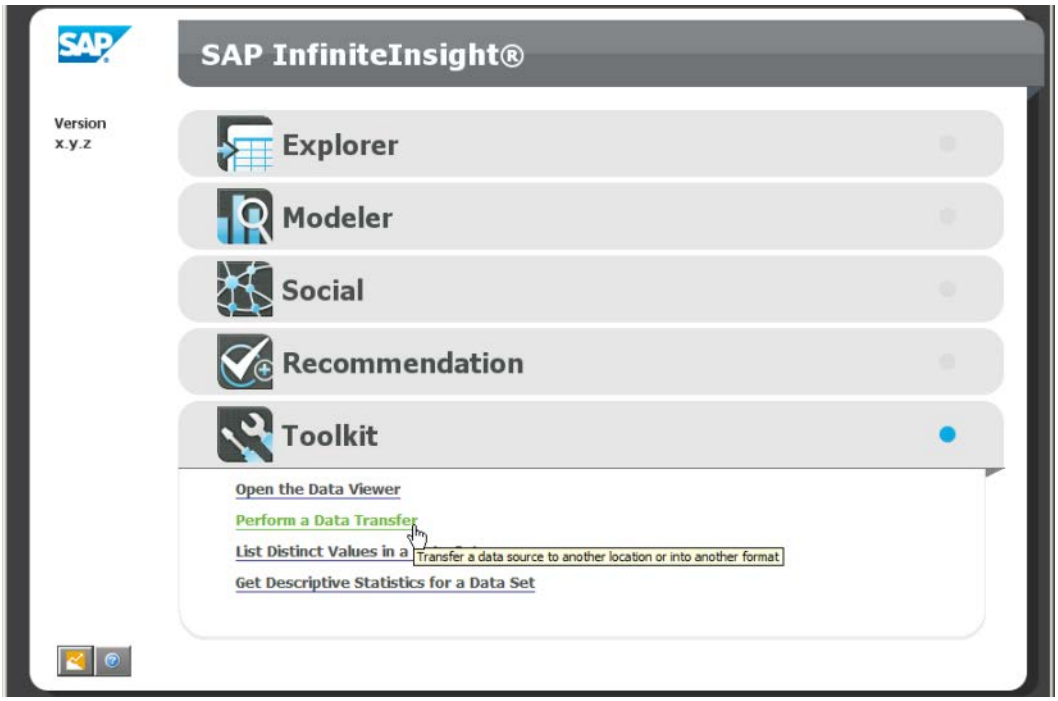
It is possible to migrate an Entity and its Time-stamped Population while performing a data transfer. This option is available via the **SAP InfiniteInsight® Toolkit** feature.

The data transfer process encompasses various steps starting from the creation of the entity and its related time-stamped population in **SAP InfiniteInsight® Explorer** to performing their transfer in the **SAP InfiniteInsight® Toolkit**.

4.1 Transfer an Entity

☑ To Transfer an Entity

1 In SAP InfiniteInsight® start panel, select the feature *Perform a Data Transfer* in the *Toolkit* section.



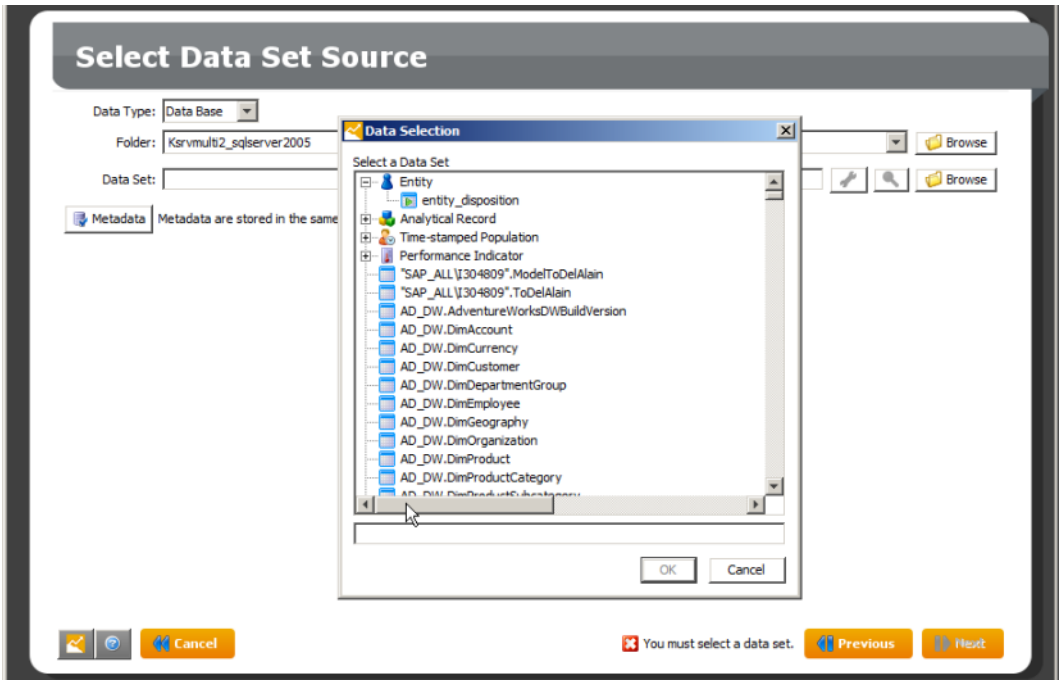
The panel *Select Data Set Source* is displayed.

2 Select:

- a *Data Type*,
- a *Folder*,

© 2014 SAP SE or an SAP affiliate company. All rights reserved-

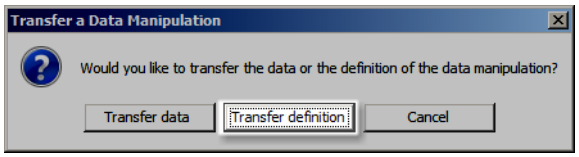
- and a *Data Set*, which is the entity you want to transfer into a new database (it is preceded by the icon with a green arrow in the middle).



3 Click *OK*.

4 Click *Next*.

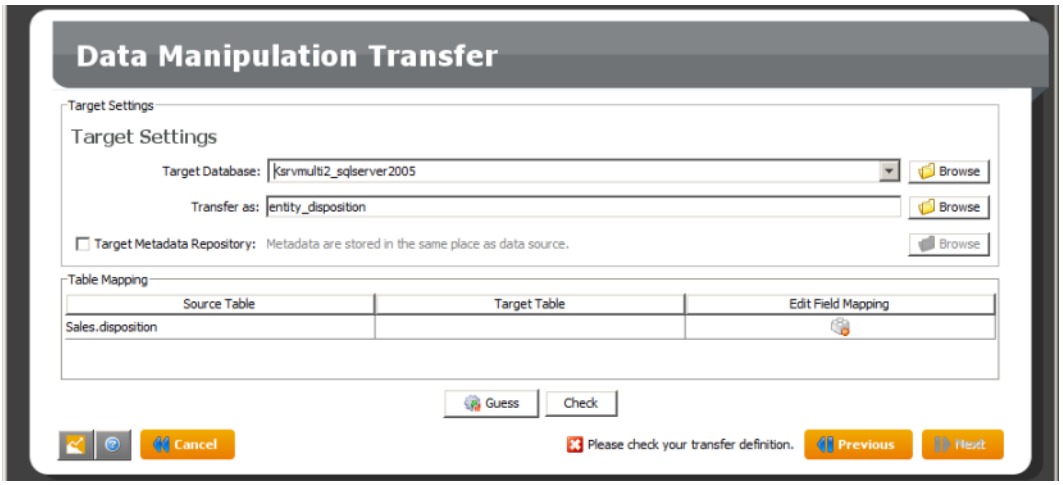
5 In the pop-up window *Transfer a Data Manipulation*, select the transfer option *Transfer Definition*.


Transfer a Data Manipulation
Would you like to transfer the data or the definition of the data manipulation?
Transfer data
Transfer definition
Cancel

The panel *Data Manipulation Transfer* appears.

6 In the area *Target Settings*:

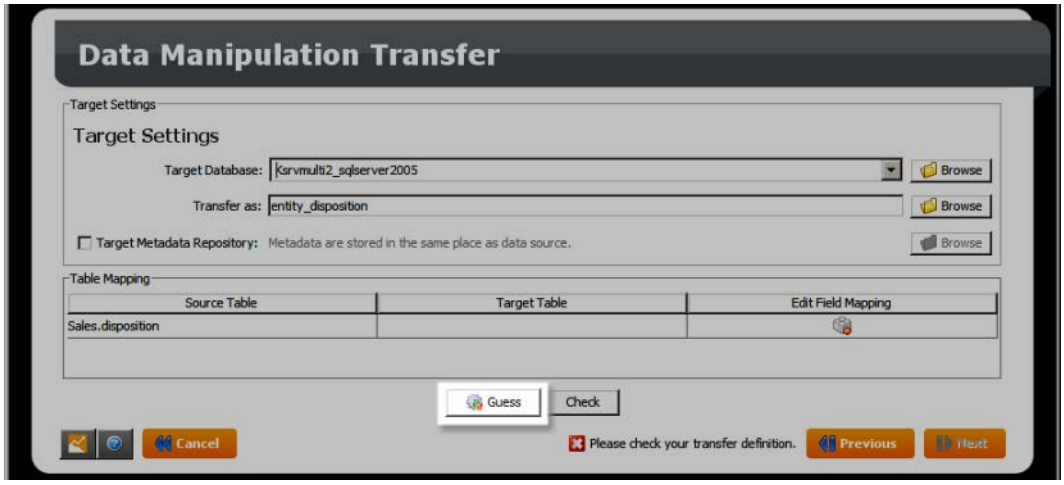
- select the *Target Database*, that is, the database where you want to transfer your data manipulation,
- in the field *Transfer as*, enter the name the data manipulation should have in the target data base.



© 2014 SAP SE or an SAP affiliate company. All rights reserved-

7 To define the data manipulation source in the area *Table Mapping*, click *Guess*.

If the new database has a similar table with a similar name, the application automatically shows the correct *Target Table*.

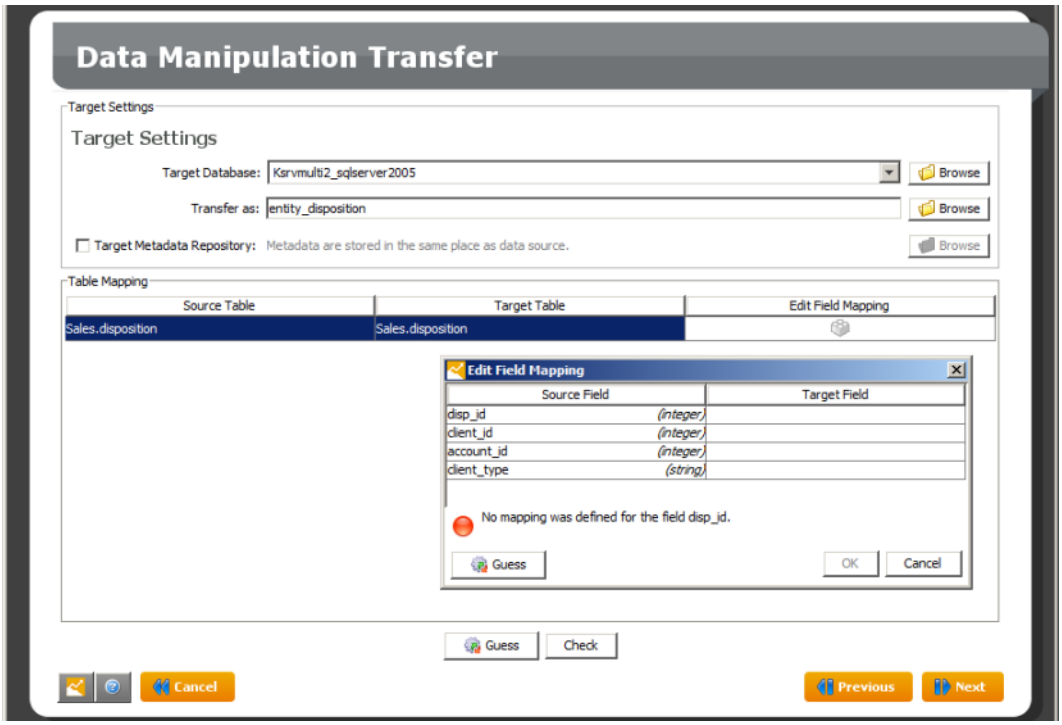


Caution

The new database must have the similar source table(s) as the existing database, that is, if the Data Manipulation is derived from sources TableA and TableB, both TableA and TableB must be migrated to the new database before performing the Data Manipulation transfer.

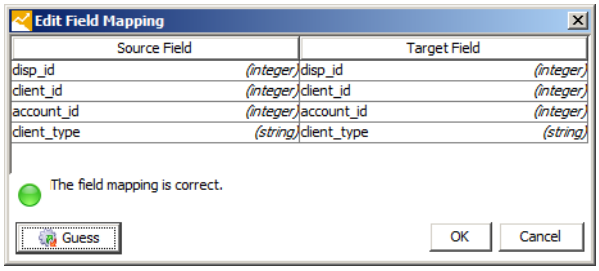
8 Edit the correct field mapping for the target by clicking the icon *Edit Field Mapping*.

The panel *Edit Field Mapping* appears. By default, there is no mapping defined.

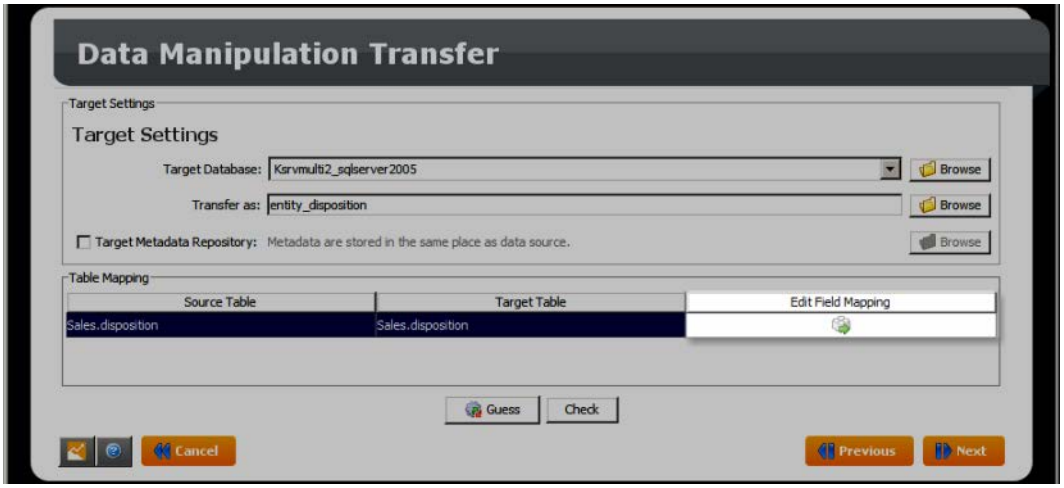


© 2014 SAP SE or an SAP affiliate company. All rights reserved-

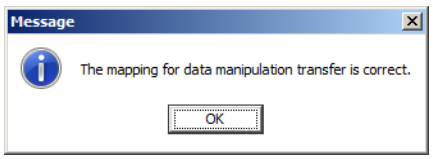
9 If you have selected a correct *Target Table* in the previous step, click *Guess.*
The *Target Fields* are automatically filled in.



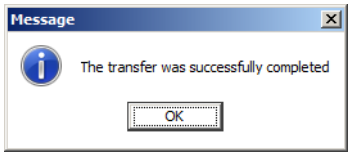
10 Click *OK*. The green arrow on the icon *Edit Field Mapping* indicates that the mapping is done.



11 Click *Check*.
A pop-window indicates that the mapping for the transfer is correct.



12 Click *OK*.
13 Click *Next*. A pop-up message indicates that the transfer has been successfully completed.
14 Click *OK*.



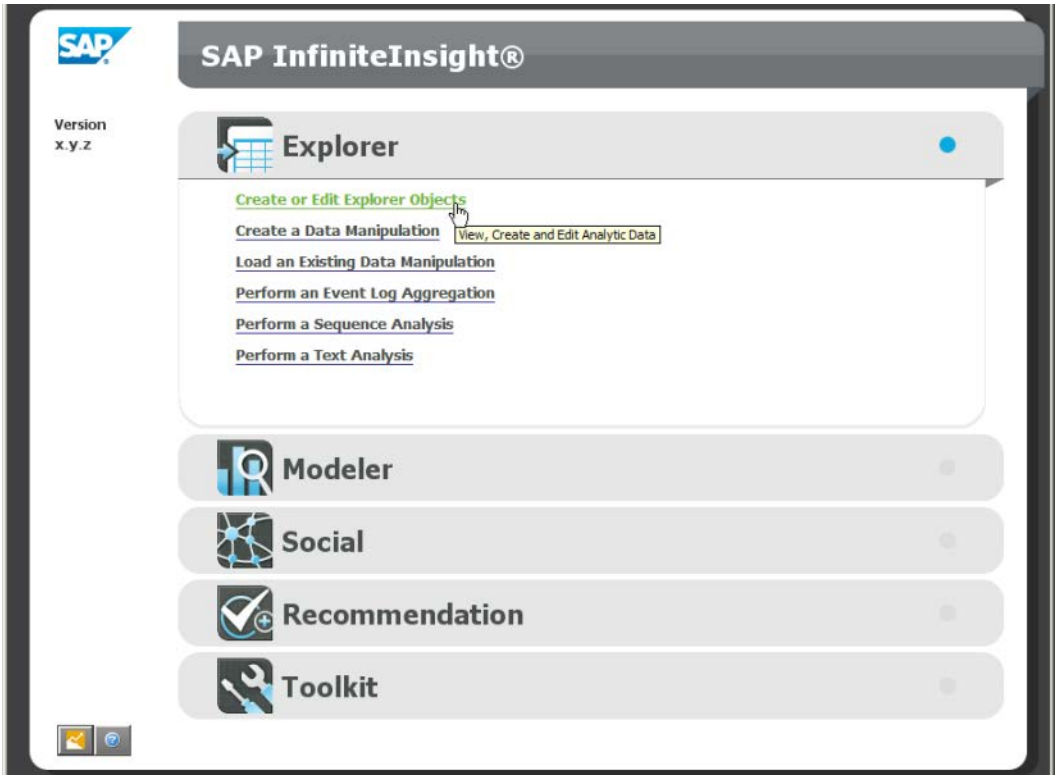
The transferred entity is now ready to be re-used in a new database.

© 2014 SAP SE or an SAP affiliate company. All rights reserved-

4.2 Use the Transferred Entity in a New Database

☑ To Use the Transferred Entity in a New Database

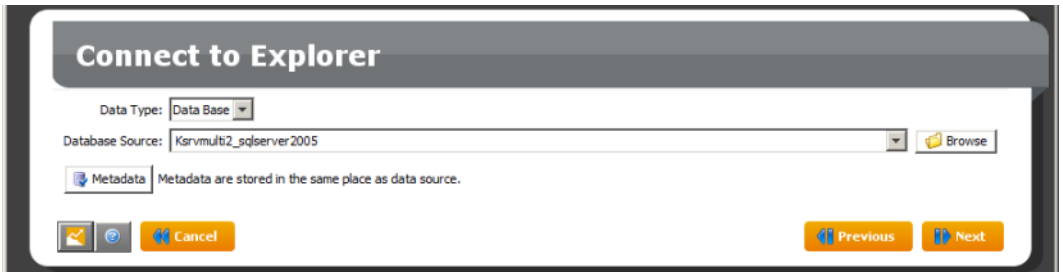
1 In the **SAP InfiniteInsight® Start Page**, click on the option *Create or Edit Explorer Objects* in the *Explorer* section.



The panel *Connect to Explorer* appears.

2 Select:

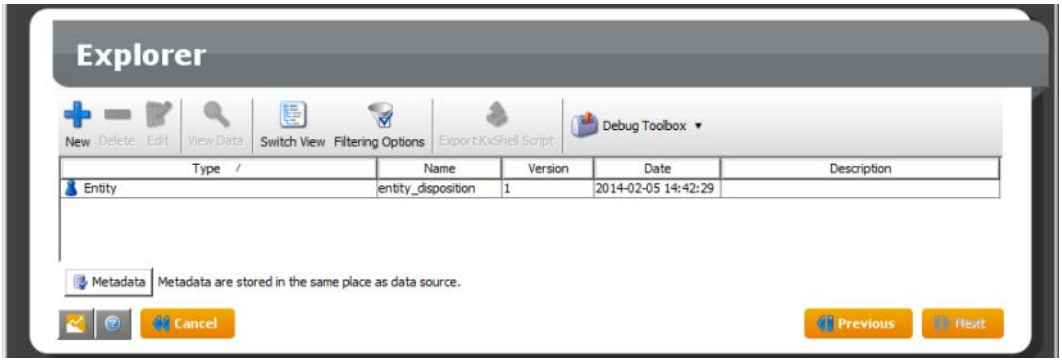
- *Data Base* in the field *Data Type*,
- and in the field *Database Source*, the database where the data manipulation has been transferred.



© 2014 SAP SE or an SAP affiliate company. All rights reserved-

3 Click *Next*.

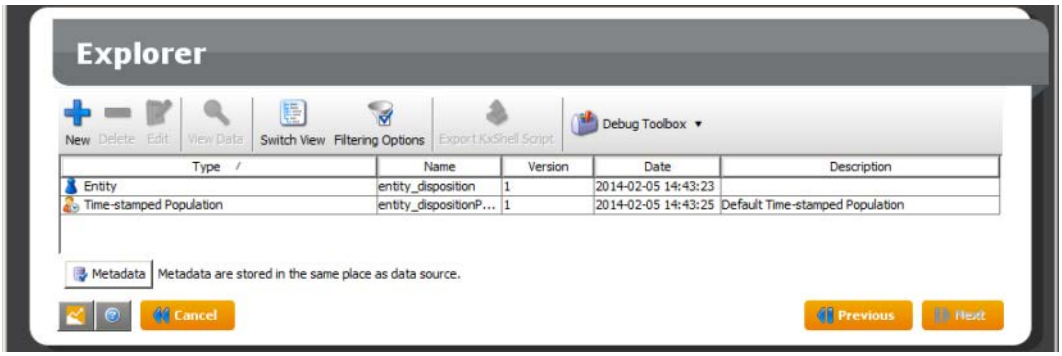
4 The transferred entity is automatically displayed.



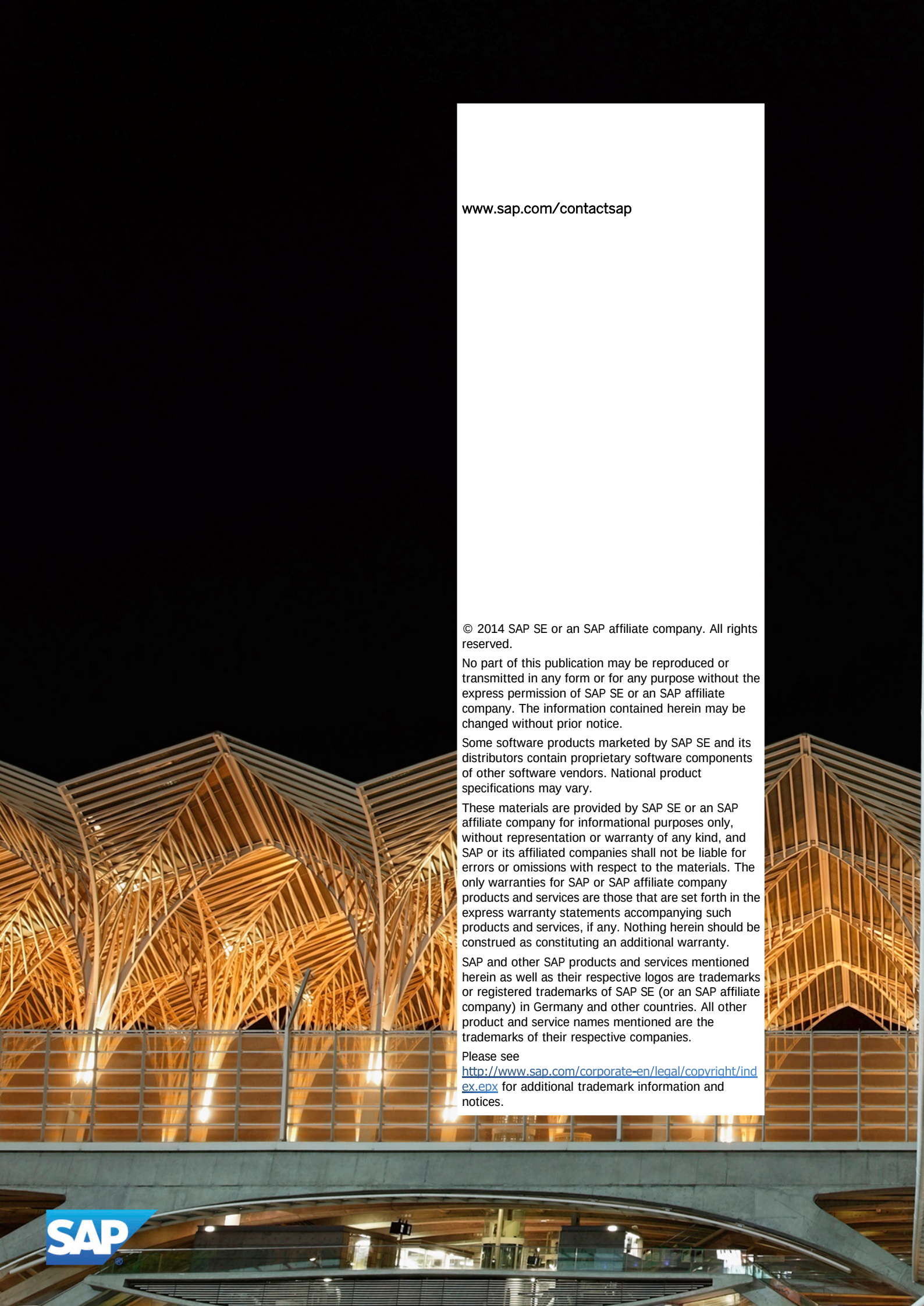
5 Click *Next* to edit the entity.

6 Perform changes and/or click *Next* to confirm.

The panel *Explorer* is displayed, listing the newly created entity and the default time-stamped population associated with it.



© 2014 SAP SE or an SAP affiliate company. All rights reserved-



www.sap.com/contactsap

© 2014 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company. The information contained herein may be changed without prior notice.

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.

Please see http://www.sap.com/corporate-en/legal/copyright/index.epx for additional trademark information and notices.